Calcul de centralité et identification de structures de communautés dans les graphes de documents

Nacim Fateh Chikhi

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par :
Université Toulouse 3 Paul Sabatier (UT3 Paul Sabatier)

Discipline ou spécialité :

Intelligence Artificielle

Présentée et soutenue par :
Nacim Fateh CHIKHI

le : vendredi 17 décembre 2010

Titre :

Calcul de centralité et identification de structures de communautés dans les graphes de documents

---

Ecole doctorale :
Mathématiques Informatique Télécommunications (MITT)

Unité de recherche :
Institut de Recherche en Informatique de Toulouse (IRIT)

Directeur(s) de Thèse :

Nathalie AUSSENAC-GILLES, Directrice de Recherche, CNRS IRIT

Rapporteurs :
Fabien GANDON, Chargé de Recherche (HDR), INRIA Sophia Antipolis
Hamamache KHEDDOUCI, Professeur, Université Claude Bernard Lyon 1

Autre(s) membre(s) du jury

Daniel GALARRETA, Ingénieur (Dr.), CNES Toulouse
Gilles RICHARD, Professeur, Université Paul Sabatier Toulouse 3
Bernard ROTHENBURGER, Ingénieur de Recherche, INRIA IRIT (Co-encadrant)

« L'homme est sage tant qu'il cherche la sagesse,
Mais dès qu'il croit l'avoir trouvée, il perd la tête. »

(Proverbe arabe)

Remerciements

En premier lieu, je tiens à remercier Bernard Rothenburger de m'avoir encadré durant toutes ces années. Il a su m'orienter, me conseiller, tout en me laissant libre de mes choix. Sa rigueur scientifique ainsi que sa disponibilité ont beaucoup contribué à l'aboutissement de ce travail de recherche.

Un grand merci à Nathalie Aussenac-Gilles, ma directrice de thèse, pour toute la confiance qu'elle m'a témoignée tout au long de cette thèse. Son expérience et son dynamisme m'ont permis de mener à bien ce travail.

Je remercie Fabien Gandon et Hamamache Kheddouci qui m'ont fait l'honneur d'être rapporteurs de ma thèse, et qui ont bien voulu me consacrer une partie de leur temps malgré leurs nombreuses occupations.

Mes remerciements vont également à Daniel Galarreta et Gilles Richard pour avoir accepté de participer au jury de ma thèse.

Je remercie tous les membres de l'équipe IC3 qui m'ont permis de réaliser ce travail dans de très bonnes conditions. Je remercie en particulier Mouna Kamel pour ses conseils et ses encouragements ainsi que Guy Camilleri pour avoir guidé mes premiers pas au sein du laboratoire IRIT lors de mon stage de Master.

En tant qu'ATER à l'Université Toulouse 1, j'ai eu le plaisir de travailler avec Chihab Hanachi et Sylvie Doutre que je tiens à remercier pour leur aide lors de l'exercice de mes fonctions.

Je tiens à remercier mon père d'avoir contribué à divers travaux de relecture, mes tantes Dalila et Hakima de m'avoir encouragé sans cesse depuis le Master et jusqu'à la fin de la thèse, sans oublier Patrick Royis pour ses remarques sur la partie qui concerne les modèles génératifs.

Durant les quelques années que j'ai passées à Toulouse, j'ai eu l'occasion de faire connaissance avec plusieurs personnes qui ont pu contribuer d'une façon ou d'une autre à ce travail et dont certaines sont devenues de très bons amis. Je pense notamment à : Ghalem, Houssem, Seifeddine, Anis, Philippe, Kader, Michel, Sami, Mouloud, Rachid, Yassine, Guillaume, Mourad, Slimane, Habib, Ali, Hichem, Lotfi, Khaldoun, ...

Je remercie beaucoup ma mère, mes tantes Dalila et Hakima, mon oncle Saadane ainsi que ma cousine Amel pour leur présence à ma soutenance.

Enfin, je remercie mes deux sœurs Amina et Imane pour leurs encouragements, et bien sûr mes parents qui m'ont toujours soutenu et à qui je dédie spécialement cette thèse.

Résumé

Dans cette thèse, nous nous intéressons à la caractérisation de grandes collections de documents (en utilisant les liens entre ces derniers) afin de faciliter leur utilisation et leur exploitation par des humains ou par des outils informatiques.

Dans un premier temps, nous avons abordé la problématique du calcul de centralité dans les graphes de documents. Nous avons décrit les principaux algorithmes de calcul de centralité existants en mettant l'accent sur le problème TKC (Tightly Knit Community) dont souffre la plupart des mesures de centralité récentes. Ensuite, nous avons proposé trois nouveaux algorithmes de calcul de centralité (MHITS, NHITS et DocRank) permettant d'affronter le phénomène TKC. Les différents algorithmes proposés ont été évalués et comparés aux approches existantes. Des critères d'évaluation ont notamment été proposés pour mesurer l'effet TKC.

Dans un deuxième temps, nous nous sommes intéressés au problème de la classification non supervisée de documents. Plus précisément, nous avons envisagé ce regroupement comme une tâche d'identification de structures de communautés (ISC) dans les graphes de documents. Nous avons décrit les principales approches d'ISC existantes en distinguant les approches basées sur un modèle génératif des approches algorithmiques ou classiques. Puis, nous avons proposé un modèle génératif (SPCE) basé sur le lissage et sur une initialisation appropriée pour l'ISC dans des graphes de faible densité. Le modèle SPCE a été évalué et validé en le comparant à d'autres approches d'ISC. Enfin, nous avons montré que le modèle SPCE pouvait être étendu pour prendre en compte simultanément les liens et les contenus des documents.

Abstract

In this thesis, we are interested in characterizing large collections of documents (using the links between them) in order to facilitate their use and exploitation by humans or by software tools.

Initially, we addressed the problem of centrality computation in document graphs. We described existing centrality algorithms by focusing on the TKC (Tightly Knit Community) problem which affects most existing centrality measures. Then, we proposed three new centrality algorithms (MHITS, NHITS and DocRank) which tackle the TKC effect. The proposed algorithms were evaluated and compared to existing approaches using several graphs and evaluation measures.

In a second step, we investigated the problem of document clustering. Specifically, we considered this clustering as a task of community structure identification (CSI) in document graphs. We described the existing CSI approaches by distinguishing those based on a generative model from the algorithmic or traditional ones. Then, we proposed a generative model (SPCE) based on smoothing and on an appropriate initialization for CSI in sparse graphs. The SPCE model was evaluated and validated by comparing it to other CSI approaches. Finally, we showed that the SPCE model can be extended to take into account simultaneously the links and content of documents.

Table des matières

Introduction 11

Motivations 11

Les graphes de documents 12

Publications dans le cadre de la thèse 14

Organisation de la thèse 15

Chapitre 1 Etat de l'art sur le calcul de centralité **..... 17**

1.1 La notion de centralité dans les graphes 18

1.2 Mesures de centralité issues de l'Analyse des Réseaux Sociaux (ARS) 20

1.2.1 Centralité de degré 20

1.2.2 Centralité de proximité 21

1.2.3 Centralité d'intermédiarité 22

1.2.4 Centralité spectrale 23

1.2.5 Limites des mesures de centralité issues de l'ARS 25

1.3 Mesures de centralité issues de la Recherche d'Information (RI) 26

1.3.1 PageRank 27

1.3.2 HITS 31

1.3.3 Salsa 34

1.3.4 HubAvg 38

1.4 Bilan 40

Chapitre 2 Nouveaux algorithmes pour le calcul de centralité dans les graphes de documents **..... 42**

2.1 L'effet TKC (Tightly Knit Community) 43

2.2 L'algorithme MHITS (Multi-HITS) 48

2.2.1 Principe 48

2.2.2 Détails de l'algorithme 49

2.2.3 Exemples jouets 50

2.2.4 Convergence de l'algorithme 50

2.3 L'algorithme NHITS (Non-negative HITS) 51

2.3.1 HITS et la décomposition en valeurs singulières 51
2.3.2 Décomposition de la matrice d'adjacence en matrices non négatives 53
2.3.3 Détails de l'algorithme 54
2.3.4 Résultats avec les graphes jouets 54
2.3.5 Effet du nombre de dimensions et convergence de l'algorithme 56
2.4 L'algorithme DocRank 56
2.4.1 Principe 56
2.4.2 Distributions stationnaires 57
2.4.3 Détails de l'algorithme 60
2.4.4 Exemples jouets 60
2.5 Environnement d'expérimentation 61
2.5.1 Graphes de documents utilisés 61
2.5.2 Mesures d'évaluation 63
2.5.3 Algorithmes comparés 64
2.6 Résultats expérimentaux 65
2.6.1 Evaluation de la qualité du classement 65
2.6.2 Evaluation de la diversité thématique 73
2.7 Bilan 74

Chapitre 3 Etat de l'art sur l'Identification de Structures de Communautés (ISC) **75**

3.1 Notion de communauté et problème de l'ISC 76
3.1.1 Définitions basées sur la connectivité des sommets 77
3.1.2 Définitions basées sur la similarité des sommets 79
3.1.3 Définitions basées sur une fonction de qualité 80
3.1.4 L'identification de structures de communautés 82
3.2 Approches non génératives pour l'ISC 83
3.2.1 Approches basées sur le clustering par partitionnement 83
3.2.2 Approches basées sur le clustering hiérarchique ascendant 85
3.2.3 Approches basées sur le clustering hiérarchique descendant 87
3.2.4 Approches basées sur le partitionnement de graphes 88
3.2.5 Autres approches 89
3.3 Approches génératives pour l'ISC 90

3.3.1 Introduction aux modèles génératifs 90
3.3.2 Approches basées sur le modèle de mélange de multinomiales 98
3.3.3 Approches basées sur le modèle PLSA 102
3.3.4 Approches basées sur le modèle SBM 110
3.4 Synthèse des approches présentées et leur adéquation à l'analyse des graphes de documents 115

Chapitre 4 Des modèles génératifs pour l'identification de structures de communautés dans les graphes de documents **120**
4.1 Le modèle SPCE (Smoothed Probabilistic Community Explorer) 121
4.1.1 Processus génératif 121
4.1.2 Estimation des paramètres 123
4.2 Mise en œuvre du modèle SPCE 127
4.2.1 Initialisation de l'algorithme EM 127
4.2.2 Estimation des paramètres de lissage 128
4.2.3 Calcul du nombre de communautés 129
4.3 Evaluation expérimentale du modèle SPCE 129
4.3.1 Evaluation de l'ISC 129
4.3.2 Effet des paramètres de lissage 132
4.3.3 Evaluation de la robustesse à la faible densité 134
4.3.4 Evaluation de la convergence 135
4.3.5 Evaluation du calcul du nombre de communautés 135
4.4 SPCE-PLSA : un modèle hybride pour l'analyse des liens et des contenus 136
4.4.1 Processus génératif 136
4.4.2 Estimation des paramètres 138
4.4.3 Mise en œuvre du modèle 140
4.4.4 Résultats expérimentaux 140
4.5 Bilan 143

Conclusion **144**

Annexe A Eléments de la théorie des graphes **147**

Annexe B Compléments au chapitre 2 ..... **149**

Bibliographie **170**

Table des Notations

Notion	Description
$\mathbf{A}$	Matrice d'adjacence d'un graphe
a_{ij}	Entrée (i,j) de la matrice $\mathbf{A}$
$\mathbf{a}_{i.}$	Vecteur correspondant à la $i^{\text{ème}}$ ligne de la matrice $\mathbf{A}$
$\mathbf{a}_{.j}$	Vecteur correspondant à la $j^{\text{ème}}$ colonne de la matrice $\mathbf{A}$
G	Un graphe $G=(V,E)$
V	Ensemble des nœuds d'un graphe
E	Ensemble des arêtes ou arcs d'un graphe
v_i	Un nœud du graphe ($v_i \in V$)
N	Nombre de sommets du graphe ($N=\lvert V\rvert$)
M	Nombre d'arêtes ou arcs du graphe ($M=\lvert V\rvert$)
$\mathbf{A}^T$	Matrice transposée de la matrice $\mathbf{A}$
$\mathbf{1}_{N\times 1}$	Vecteur colonne de dimension N contenant des uns
$\mathbf{0}_{N\times 1}$	Vecteur colonne de dimension N contenant des zéros
$\lVert\mathbf{x}\rVert_1$	Norme L1 du vecteur $\mathbf{x}$ ($\lVert\mathbf{x}\rVert_1=\sum_{i=1} x_i$)
$\lVert\mathbf{x}\rVert_2$	Norme L2 du vecteur $\mathbf{x}$ ($\lVert\mathbf{x}\rVert_2=\sqrt{\sum_{i=1} x_i^2}$)
d_i	Degré total du nœud v_i (nombre total de liens)
d_i^{in}	Degré entrant du nœud v_i (nombre de liens entrants)
d_i^{out}	Degré sortant du nœud v_i (nombre de liens sortants)
K	Nombre de communautés
$x\sim \text{Mult}(k,\mathbf{p})$	k tirages suivant une loi multinomiale de paramètre $\mathbf{p}$
$x\sim \text{Bern}(p)$	Tirage selon une loi de Bernoulli de paramètre p
$x\sim \text{Dir}(\alpha)$	Tirage selon une loi de Dirichlet de paramètre α

Introduction

Motivations

L'infobésité (ou surcharge d'information) constitue sans doute l'un des plus grands défis auxquels nous sommes confrontés aujourd'hui. Que nous soyons industriels, chercheurs scientifiques ou simples utilisateurs du web, nous faisons tous face à cette immense quantité d'information disponible dont l'appréhension, l'ampleur, le contenu (et donc l'accès) nous échappent. Même une cartographie de l'existant dépasse nos capacités de traitement cognitif. Certaines avancées technologiques accentuent même ce phénomène à l'instar du Web 2.0 qui permet à n'importe quel internaute de déposer facilement et rapidement de l'information sur le web. Devant une telle situation, il devient plus que jamais nécessaire de disposer de moyens permettant de caractériser ces ressources documentaires afin d'en faciliter l'accès. La caractérisation peut porter sur les sources documentaires elles-mêmes, prises séparément, comme le proposent le Web 2.0 ou même le web sémantique. Mais on peut aussi vouloir caractériser globalement ces grandes quantités de ressources, comme un tout auquel on peut attribuer des propriétés. C'est ce dernier aspect qui nous intéresse dans cette thèse. Deux critères peuvent être envisagés pour caractériser ces grandes collections de documents.

Le premier de ces critères est celui de la (plus ou moins grande) pertinence que l'on accordera à un document pour répondre à une interrogation donnée (i.e. à une requête). La solution la plus naturelle pour estimer cette pertinence est de mesurer la proximité entre le contenu d'un document et le contenu de la requête. Une solution radicalement différente consiste à caractériser a priori l'importance des différents documents [Getoor and Diehl 05]. Pour estimer cette importance, on a coutume de considérer qu'un document est d'autant plus important qu'il existe un grand nombre de documents qui pointent vers lui [Garfield 70],[Liu 06]. Si l'on considère le graphe des liens entre documents (que l'on appellera graphe de documents tout court), on s'aperçoit que les documents importants sont alors ceux qui

occupent des positions *centrales* (ou stratégiques) dans un tel graphe. Dans le développement de cette thèse, nous privilégierons cette deuxième solution.

Le second de ces critères vient de la capacité à structurer de grandes collections de documents en groupes traitant chacun d'un même sujet ou d'une même thématique. Là encore, la majorité des techniques existantes s'est focalisée sur l'utilisation des contenus textuels. Le regroupement obtenu correspond alors à des ensembles de documents qui sont à la fois fortement similaires au sein d'un même groupe et faiblement similaires aux documents d'autres groupes. Cependant, dans le cadre de cette thèse, nous nous intéresserons à l'utilisation des liens entre documents pour le regroupement automatique de documents. Les liens entre documents ont été utilisés depuis longtemps en bibliométrie pour l'évaluation des revues scientifiques par exemple. La plupart des techniques existantes basées sur les liens se contentent, en fait, d'utiliser d'anciennes notions issues de la bibliométrie telles que la co-citation [Small 73] ou le couplage bibliographique [Kessler 63]. Nous pensons pourtant que les liens contribuent à la sémantique d'un document, et qu'ils constituent par conséquent une source d'information précieuse qu'il serait important de prendre en compte lors de la tâche d'identification de thématiques. Concrètement, le regroupement obtenu en se basant sur les liens correspond à des ensembles de documents qui ont plus de liens à l'intérieur qu'à l'extérieur de chacun des groupes [Getoor and Diehl 05]. En observant le graphe de liens entre documents, on s'aperçoit que ces groupes de documents sont ceux qui partagent des préoccupations communes, on dit qu'ils constituent des *communautés* à l'intérieur du graphe.

Ces deux critères méritent d'être distingués. Le premier est sensé se focaliser sur quelques documents qui l'emportent, toutes thématiques confondues. Le second concerne tous les documents en les classant selon des thématiques particulières. Pour autant, ces deux critères ne sont pas indépendants. Le calcul de documents centraux dans des graphes de documents est souvent biaisé par la présence de communautés dans le graphe. Inversement, une fois des communautés identifiées, il est nécessaire d'identifier les éléments centraux de chacune d'elles afin de faciliter leur interprétation.

Les graphes de documents

Les objets que nous manipulerons tout au long de cette thèse sont des graphes de documents. Un graphe de documents est un graphe où les sommets correspondent à des documents et les liens à des références créées par le(s) auteur(s) des documents. Des exemples de graphes de documents incluent les graphes de citations (i.e. d'articles scientifiques reliés par des références bibliographiques) [Price 65][Scharnhorst and Thelwall 05], les graphes du web (i.e. de pages web reliées par des hyperliens) [Kleinberg 99b][Scharnhorst and Thelwall 05] et les graphes de brevets [Yoon and Park 04].

De nombreuses études empiriques réalisées ces dernières années sur des graphes dans divers domaines (biologie, sciences sociales, web, etc.) ont montré que ces graphes possèdent des propriétés communes qui les distinguent des graphes aléatoires tels que les graphes d'Erdös-Rényi [Fortunato 10]. Ces graphes "non aléatoires" sont connus sous les noms de réseaux complexes [Bornholdt and Schuster 03], de graphes d'interaction [Guillaume 04] ou encore de grands graphes de terrain [Pons 07]. Les graphes de documents sont justement un

cas particulier de graphes complexes ayant des particularités. Une des caractéristiques les plus évidentes concerne l'*orientation* des liens dans un graphe de documents. En effet, alors que la plupart des graphes complexes étudiés sont non-orientés, les graphes de documents sont orientés. Cette orientation possède une sémantique bien précise qu'il serait incorrect de négliger. En plus de l'orientation, nous présentons ci-dessous trois des propriétés les plus importantes des graphes de documents.

a) Distribution des degrés en loi de puissance

Une caractéristique importante des graphes de documents est que la distribution des degrés de leurs nœuds (i.e. du nombre de liens) ne suit pas une loi de Poisson comme c'est le cas des graphes aléatoires d'Erdös-Rényi, mais suit plutôt une loi de puissance [Redner 98][Albert et al. 99][Broder et al. 00]. Cette particularité a en effet été rapportée par de nombreuses études empiriques qui se sont intéressées à la distribution des degrés dans des graphes de documents. La loi de puissance énonce que la probabilité, *P(k)*, qu'un nœud ait un degré égal exactement à *k*, est donnée par [Caldarelli 07] :

$$P(k) = Z.k^{-\lambda}$$

où *Z* est une constante de normalisation et λ, l'exposant de la loi, est un réel dont la valeur est généralement comprise entre 2 et 3. Cette loi indique donc que le nombre de nœuds ayant un degré égal à *k* est proportionnel à $k^{-\lambda}$. Elle traduit également le fait que dans les réseaux complexes, un petit nombre (non négligeable) de nœuds possède un fort degré alors qu'un grand nombre de nœuds possède un faible degré.

Une autre interprétation de la distribution des degrés en loi de puissance est que, dans un graphe, les nœuds jouent des rôles différents. Il est alors particulièrement intéressant de repérer les nœuds qui occupent une position stratégique dans le graphe. La Partie I de cette thèse est d'ailleurs consacrée à cette problématique.

Notons enfin que les graphes possédant une telle propriété (i.e. distribution des degrés en loi de puissance) sont appelés *graphes sans échelle* (ou graphes à invariance d'échelle ou encore scale-free graphs) [Barabasi 09].

b) Faible degré moyen et faible densité

Une autre propriété des graphes de documents concerne le faible degré moyen [Albert et al. 99][Broder et al. 00]. Pour un graphe orienté et sans boucles $G = (V, E)$ d'ordre *N* et de taille *M*, le degré moyen est donné par :

$$d^{moy} = \frac{M}{N}$$

Le fait que les graphes de documents aient un degré moyen faible implique que l'on peut considérer que le nombre de liens dans un graphe est linéaire (i.e. du même ordre) par rapport au nombre de nœuds.

La densité d'un graphe est égale à la proportion de liens existants par rapport au nombre total de liens possibles. Pour le graphe orienté et sans boucles G, sa densité δ est donnée par :

$$\delta = \frac{M}{N(N-1)}$$

Le nombre de liens M étant du même ordre que le nombre de nœuds N, la densité δ tend par conséquent vers des valeurs très faibles lorsque le nombre de nœuds croit. En d'autres termes, la densité d'un graphe est inversement proportionnelle au nombre de ses sommets.

Une appellation souvent utilisée pour désigner les graphes ayant une faible densité est celle de *graphes creux* (ou "sparse") en référence à leur matrice d'adjacence qui est creuse (i.e. contient beaucoup de zéros).

c) Fort coefficient de regroupement

Le coefficient de regroupement appelé aussi coefficient de clustering exprime la probabilité que les voisins d'un nœud soient eux-mêmes connectés par un lien [Caldarelli 07]. Formellement, dans un graphe orienté $G=(V,E)$ d'ordre N représenté par sa matrice d'adjacence **A**, le coefficient de regroupement d'un nœud i est défini par :

$$C_i = \frac{\sum_{j \in V_i} \sum_{k \in V_i} a_{jk}}{d_i(d_i-1)}$$

où d_i est le degré total du nœud i (degré entrant + degré sortant), $V_i = \{ j : a_{ij} = 1 \vee a_{ji} = 1 \}$ est l'ensemble des nœuds voisins du nœud i.

Le coefficient de regroupement global d'un graphe est égal à la moyenne des coefficients de regroupement de tous ses nœuds i.e.

$$C(G) = \frac{1}{N} \sum_{i=1}^{N} C_i$$

Les graphes de documents sont caractérisés par un fort coefficient de regroupement [Baldi and al. 03]. Cette forte valeur du coefficient de regroupement d'un graphe indique la présence d'ensembles de nœuds fortement connectés entre eux localement et ce en dépit de la faible densité globale du graphe. Intuitivement, cela correspond à la présence d'une structure de communautés dans le graphe [Porter et al. 09]. L'identification de telles structures fait l'objet de la Partie II de cette thèse.

Publications dans le cadre de la thèse

a) Conférences internationales avec comité de lecture

- N. F. Chikhi, B. Rothenburger and N. Aussenac-Gilles. Community Structure Identification: A Probabilistic Approach. In *International Conference on Machine Learning and Applications (ICMLA)*, Miami, Florida (USA), IEEE Computer Society, pages 125-130, 2009 (Best paper award).

- N. F. Chikhi, B. Rothenburger and N. Aussenac-Gilles. Combining Link and Content Information for Scientific Topics Discovery. In *IEEE International Conference on Tools with Artificial Intelligence (ICTAI)*, Dayton, Ohio (USA), IEEE Computer Society, pages 211-214, 2008.
- N. F. Chikhi, B. Rothenburger and N. Aussenac-Gilles. Authoritative Documents Identification based on Nonnegative Matrix Factorization. In *IEEE International Conference on Information Reuse and Integration (IRI)*, Las Vegas, Nevada (USA), IEEE, pages 262-267, 2008.
- N. F. Chikhi, B. Rothenburger and N. Aussenac-Gilles. A New Algorithm for Community Identification in Linked Data. In *International Conference on Knowledge-Based and Intelligent Information & Engineering Systems (KES)*, Zagreb (Croatia), Springer-Verlag, pages 641-648, 2008.
- N. F. Chikhi, B. Rothenburger and N. Aussenac-Gilles. A Comparison of Dimensionality Reduction Techniques for Web Structure Mining. In *IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, Silicon Valley, California (USA), IEEE Computer Society, pages 116-119, 2007.

b) Conférences nationales avec comité de lecture

- N. F. Chikhi, B. Rothenburger and N. Aussenac-Gilles. Une approche probabiliste pour l'identification de structures de communautés. In *Conférence Internationale Francophone sur l'Extraction et la Gestion des Connaissances (EGC)*, Hammamet (Tunisie), Cépaduès Editions, pages 175-180, 2010.

c) Workshops nationaux avec comité de lecture

- B. Rothenburger, N. F. Chikhi and N. Aussenac-Gilles. Le web sémantique est-il soluble dans le web 2.0? (Fouille de texte versus Fouille de communauté). In *Atelier IC2.0, en association avec les 19èmes journées Francophones d'Ingénierie des Connaissances (IC2008)*, Nancy (France), INRIA, pages 17-18, 2008.
- B. Rothenburger, N. F. Chikhi and N. Aussenac-Gilles. Comment maintenir l'intelligibilité des archives scientifiques : une solution basée sur l'utilisation des ontologies et des textes. In *Atelier Ontologies et Textes, en association avec la conférence Terminologie et Intelligence Artificielle (TIA07)*, Sophia-Antipolis (France), 2007.

Organisation de la thèse

Cette thèse est organisée en deux parties qui peuvent être lues de manière indépendante. La première partie est consacrée au calcul de centralité dans les graphes de documents tandis que la deuxième partie traite la problématique de l'identification de structures de communautés (ISC) dans les graphes de documents. Chacune de ces deux parties est à son tour composée de deux chapitres : un chapitre d'état de l'art critique et un chapitre qui présente les nouvelles approches proposées dans cette thèse ainsi que leur évaluation.

Le chapitre 1 aborde la problématique du calcul de centralité dans les graphes. Il commence par donner une intuition de la notion de centralité. Il décrit ensuite les mesures de centralité proposées dans le cadre de l'analyse de réseaux sociaux à savoir la centralité de degré, la centralité d'intermédiarité, la centralité de proximité et la centralité spectrale. L'applicabilité de ces mesures avec des graphes de documents est discutée. Des algorithmes de calcul de centralité issus de la recherche d'information sont ensuite décrits de manière détaillée. Il s'agit des algorithmes PageRank, HITS, Salsa et HubAvg. Leurs avantages et inconvénients sont également mentionnés.

Le chapitre 2 décrit tout d'abord les causes de l'effet TKC (Tightly Knit Community) qui caractérise un grand nombre d'algorithmes de calcul de centralité. Il présente ensuite les trois nouveaux algorithmes de calcul de centralité à savoir MHITS (Multi-HITS), NHITS (Non-negative HITS) et DocRank. Enfin, une partie expérimentale est proposée dans laquelle nos algorithmes sont comparés aux algorithmes existants en utilisant plusieurs graphes de documents ainsi que plusieurs mesures d'évaluation.

Les différentes définitions de la notion de communauté sont données au début du chapitre 3. Nous présentons ensuite les différentes approches existant pour l'identification de structures de communautés dans les graphes, en distinguant les approches génératives des approches non génératives. Les premières recevront une attention particulière car les modèles d'ISC proposés dans cette thèse sont basés sur ce principe. Nous décrivons notamment les méthodes basées sur le modèle de mélange de multinomiales, sur le modèle PLSA (Probabilistic Latent Semantic Analysis) ou sur le modèle SBM (Stochastic Block Model).

Le chapitre 4 regroupe le deuxième ensemble de nos contributions. Il décrit le modèle génératif SPCE (Smoothed Probabilistic Community Explorer) que nous proposons pour l'ISC afin de résoudre le problème de la faible densité des graphes de documents. Différents aspects concernant la mise en œuvre du modèle SPCE (initialisation, calcul du nombre de communautés, etc.) sont notamment présentés. Ensuite, une étude expérimentale évalue et compare notre modèle à d'autres algorithmes existants d'ISC. Enfin, un modèle génératif hybride basé à la fois sur le modèle SPCE pour l'analyse des liens, et sur le modèle PLSA pour l'analyse des contenus est proposé et évalué pour le regroupement automatique de documents.

Finalement, les annexes présentent un rappel sur la théorie des graphes ainsi qu'un complément des résultats du chapitre 2.

1

Etat de l'Art sur le Calcul de Centralité

Ce chapitre vise à donner une vue d'ensemble sur le calcul de centralité dans les graphes en général puis à présenter de manière détaillée divers algorithmes de calcul de centralité dans les graphes de documents. Nous commençons par présenter diverses mesures de centralité proposées par les chercheurs en science sociales dans le cadre de l'analyse des réseaux sociaux (ARS). Ces mesures incluent la centralité de degré, de proximité, d'intermédiarité et de vecteur propre. Nous montrons ensuite les raisons pour lesquelles ces mesures dites classiques ne sont pas adaptées à l'analyse de graphes de documents.

Les graphes de documents possèdent des particularités qui ont nécessité le développement de mesures de centralité spécifiques. Ces mesures ont été principalement proposées en recherche d'information (RI) dans le but d'améliorer la performance des moteurs de recherche sur le web. Nous décrivons quelques uns des algorithmes les plus emblématiques pour le calcul de degrés d'importance dans les graphes de documents notamment l'algorithme PageRank de Brin et Page et l'algorithme HITS de Kleinberg. Les avantages et inconvénients de chacun des algorithmes sont également exposés en insistant sur un problème récurrent à savoir l'effet TKC (Tightly Knit Community) dont souffre un grand nombre de ces approches.

1.1 La notion de centralité dans les graphes

Dans le cadre de l'analyse des réseaux sociaux, les chercheurs ont remarqué que certains acteurs jouent un rôle plus "important" que d'autres. Certaines personnes avaient par exemple beaucoup de contacts au sein du réseau alors que d'autres personnes n'en avaient que très peu. Ces personnes occupent en fait une position stratégique (ou avantageuse) au sein du réseau social ; suivant les cas, ces personnes peuvent être plus influentes ou avoir une plus grande notoriété. Ce phénomène n'est pas propre aux réseaux sociaux puisqu'on le retrouve dans beaucoup d'autres types de graphes. Par exemple, dans les réseaux de communication, certains nœuds jouent un rôle très important dans la communication entre les nœuds alors que d'autres nœuds n'ont aucune influence sur le réseau. Ce phénomène s'explique par la nature de ces réseaux qui sont des réseaux complexes et dont la distribution des nœuds suit une loi de puissance [Caldarelli 07]. Cette loi de probabilité signifie qu'un petit nombre de nœuds possède beaucoup de connexions (i.e. de liens) alors qu'un grand nombre de nœuds possède peu de liens avec les autres nœuds du réseau. En d'autres termes, les nœuds n'ont pas les mêmes rôles au sein du réseau.

Dans le but de quantifier cette notion d'importance d'un nœud dans un graphe, les chercheurs ont proposé plusieurs définitions connues sous le nom de *mesures de centralité* [Koschützki et al. 05]. En effet, l'identification des nœuds centraux dans un graphe représente un enjeu important dans plusieurs domaines. En reprenant l'exemple des réseaux de communication, le fait de connaître les nœuds importants permet d'adopter des stratégies afin de mieux protéger ces nœuds qui jouent un rôle essentiel dans la communication au sein du réseau. Si l'on considère à titre d'exemple le réseau de communication de la figure 1.1, on remarque que si le nœud B tombe en panne, la communication entre les différents nœuds ne sera pas affectée. Par contre, si le nœud A tombe en panne, cela engendrerait un grand problème de communication puisque tous les nœuds se retrouveront isolés.

En recherche d'information, le classement (ou "ranking") des résultats est basé non seulement sur une mesure de similarité entre la requête utilisateur et les documents mais aussi sur le degré d'importance (ou d'autorité) d'un document dans le graphe de citations.

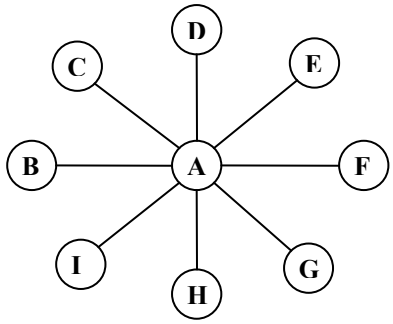


Figure 1.1 – Exemple d'un réseau de communications

Lors de l'étude de la notion de centralité, il est important de distinguer le cas des graphes orientés du cas des graphes non-orientés. Dans les graphes orientés, les nœuds possèdent deux types de liens, à savoir des liens entrants et des liens sortants. Pour une définition donnée de la notion de centralité, chaque nœud aura alors deux mesures d'importance : une mesure relative à ses liens sortants, appelée mesure de *centralité, d'influence, d'hubité ou de centralité sortante*, et une autre mesure relative à ses liens entrants, appelée mesure de *prestige, de popularité, d'autorité ou de centralité entrante* [Wasserman and Faust 94]. Dans un graphe non-orienté, chaque nœud possède un seul type de liens ou de relations avec les autres nœuds. Chaque nœud possède alors une seule mesure d'importance (correspondant à une définition précise) appelée mesure de *centralité* [Wasserman and Faust 94].

Dans la suite de ce chapitre, nous décrivons quelques mesures de centralité dans les graphes orientés et non-orientés. Nous commençons par rappeler les principales mesures proposées dans le domaine de l'analyse des réseaux sociaux. Celles-ci incluent les centralités de degré, d'intermédiarité, de proximité ainsi que la centralité du vecteur propre (appelée aussi centralité spectrale). Ensuite, nous présentons quelques uns des algorithmes les plus populaires pour le calcul de l'importance d'un document dans un graphe d'hyperliens ou de références bibliographiques. Il s'agit des algorithmes PageRank, HITS, Salsa et HubAvg. Afin d'illustrer les différentes mesures de centralité, nous utiliserons les trois graphes jouets de la figure 1.2

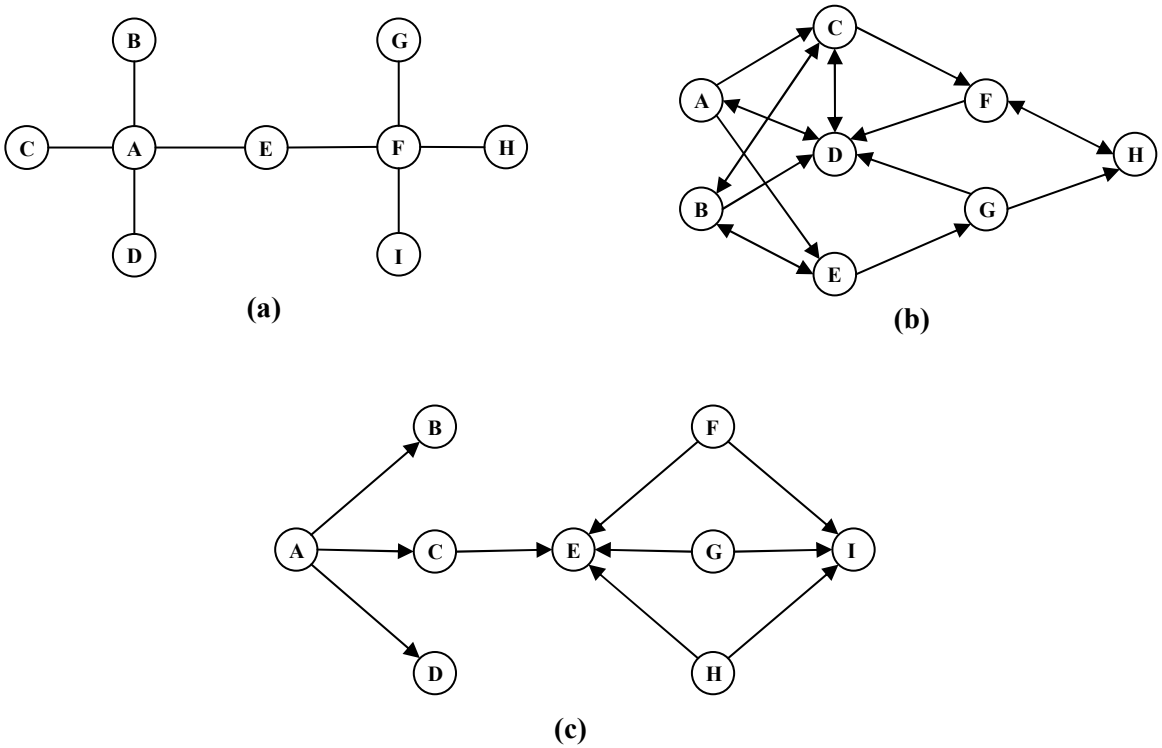


Figure 1.2 – (a) Graphe jouet *G1* ; (b) Graphe jouet *G2* ; (c) Graphe jouet *G3*

1.2 Mesures de centralité issues de l'Analyse des Réseaux Sociaux (ARS)

Le calcul de centralité est depuis plusieurs décennies une problématique importante dans le domaine de l'analyse des réseaux sociaux [Wasserman and Faust 94]. La centralité est une notion qui permet de rendre compte de la popularité ou visibilité d'un acteur au sein d'un groupe. L'article "*Centrality in social networks: Conceptual clarification [Freeman 79]*" de Freeman représente sans doute l'une des contributions les plus importantes dans le domaine de l'analyse des réseaux sociaux. Dans son article, Freeman propose trois définitions formelles du concept de centralité que nous présentons ci-dessous. Nous présentons également une quatrième définition introduite par Bonacich qui est à la base d'un grand nombre de mesures de centralité dans les graphes de documents.

1.2.1 Centralité de degré

La centralité de degré [Freeman 79] représente la forme la plus simple et la plus intuitive de la notion de centralité. Elle est basée sur l'idée que l'importance d'un individu au sein d'un groupe dépend du nombre total de personnes qu'il connaît ou avec lesquelles il interagit directement. Selon cette mesure, déterminer l'importance d'un nœud dans un graphe revient donc à calculer le nombre de ses sommets voisins, ou de manière équivalente, à calculer le nombre de liens qui lui sont incidents. En théorie des graphes, ce nombre est appelé *degré* du nœud, d'où l'appellation de centralité de degré.

Soit $G=(V,E)$ un graphe d'ordre N représenté par sa matrice d'adjacence $\mathbf{A}$. Dans le cas où le graphe G est non-orienté, la centralité de degré d'un nœud $v_i \in V$ est définie par :

$$C^{\deg}(v_i) = \frac{1}{N-1}\sum_{j=1}^{N} a_{ij} \tag{1.1}$$

En notation matricielle, le vecteur de la centralité de degré est donné par : $\mathbf{c}^{\deg} = \frac{\mathbf{A}\times\mathbf{1}}{N-1}$, où $\mathbf{1}$ est un vecteur colonne de dimension N contenant des uns.

Dans le cas où le graphe G est orienté, chaque nœud $v_i \in V$ possède alors deux mesures de centralité de degré : une par rapport aux liens sortants et une par rapport aux liens entrants. Elles sont définies respectivement par :

$$C_{out}^{\deg}(v_i) = \frac{1}{N-1}\sum_{j=1}^{N} a_{ij} \;\; ; \;\; C_{in}^{\deg}(v_i) = \frac{1}{N-1}\sum_{j=1}^{N} a_{ji}$$

En termes de matrices, les vecteurs de centralité de degré sortant et de degré entrant sont donnés respectivement par $\mathbf{c}_{out}^{\deg} = \frac{\mathbf{A}\times\mathbf{1}}{N-1}$ et $\mathbf{c}_{in}^{\deg} = \frac{\mathbf{A}^T\times\mathbf{1}}{N-1}$.

Les figures 1.3 et 1.4 indiquent respectivement la centralité de degré pour les nœuds graphes jouets *G1* et *G2*. Les résultats fournis par cette analyse de la centralité de degré montrent que les nœuds A et F, ayant quatre liens chacun, sont les plus importants dans le

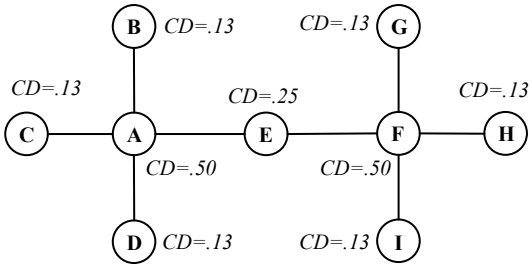


Figure 1.3 - Centralité de degré (CD) pour les nœuds du graphe *G1*

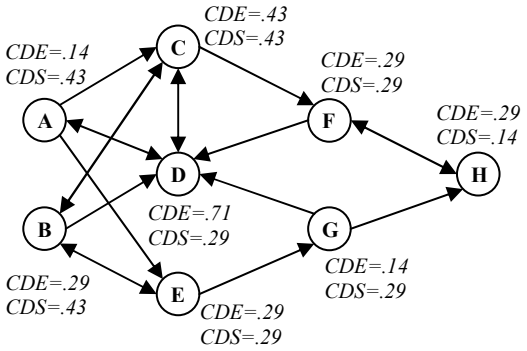


Figure 1.4 - Centralités de degré entrant (CDE) et sortant (CDS) pour les nœuds du graphe *G2*

graphe *G1*. Pour le graphe *G2*, les nœuds A, B et C possèdent la plus forte centralité par rapport aux liens sortants, tandis que le nœud D possède la plus forte centralité par rapport aux liens entrants.

La centralité de degré est aussi appelée mesure de centralité locale [Scott 00] car elle ne prend pas en compte la structure globale du graphe et n'est calculée qu'à partir du voisinage immédiat d'un sommet. Bien qu'elle soit pertinente dans certaines situations, la centralité de degré s'avère être peu informative dans d'autres cas, comme par exemple pour l'analyse des graphes de pages web [Kleinberg 99a]. Les mesures que nous présentons dans la suite sont toutes des mesures de centralité globales.

1.2.2 Centralité de proximité

La centralité de proximité [Freeman 79] est une mesure de centralité globale basée sur l'intuition qu'un nœud occupe une position stratégique (ou avantageuse) dans un graphe s'il est globalement proche des autres nœuds de ce graphe. Par exemple dans un réseau social, cette mesure correspond à l'idée qu'un acteur est important s'il est capable de contacter facilement un grand nombre d'acteurs avec un minimum d'effort (l'effort ici est relatif à la taille des chemins). En pratique, la centralité de proximité d'un nœud est obtenue en calculant sa proximité moyenne vis-à-vis des autres nœuds du graphe.

Soit $G=(V,E)$ un graphe d'ordre N représenté par sa matrice d'adjacence $\mathbf{A}$. Dans le cas où le graphe G est non-orienté, la centralité de proximité d'un nœud $v_i \in V$ est définie par :

$$C^{pro}(v_i) = \frac{N-1}{\sum_{j=1}^{N} dist(v_i, v_j)} \tag{1.2}$$

où $dist(v_i, v_j)$ est la distance entre les deux sommets v_i et v_j.

Dans le cas où le graphe G est orienté, chaque nœud $v_i \in V$ possède alors deux mesures de centralité de proximité : une par rapport aux liens sortants et une par rapport aux liens entrants. Elles sont définies respectivement par :

$$C_{out}^{pro}(v_i) = \frac{N-1}{\sum_{j=1}^{n} dist(v_i, v_j)} \quad ; \quad C_{in}^{pro}(v_i) = \frac{N-1}{\sum_{j=1}^{n} dist(v_j, v_i)}$$

où $dist(v_i, v_j)$ est la distance entre les deux sommets v_i et v_j.

Pour le calcul des distances entre sommets, plusieurs métriques peuvent être utilisées. Freeman propose par exemple d'utiliser la distance géodésique (i.e. taille du chemin le plus court) entre les nœuds. D'autres mesures de distance telle que la distance euclidienne peuvent également être utilisées pour le calcul de la centralité de proximité.

Les figures 1.5 et 1.6 indiquent respectivement la centralité de proximité (de Freeman) pour les nœuds des graphes jouets *G1* et *G2*. On remarque que le nœud E est le plus important dans le graphe *G1* alors qu'il n'a que deux liens. Cela est dû au fait que le nœud E est le moins excentré dans le graphe. Pour le graphe *G2*, on obtient des résultats proches de ceux de la centralité de degré i.e. les nœuds A et B possèdent la plus forte centralité par rapport aux liens sortants, tandis que le nœud D possède la plus forte centralité par rapport aux liens entrants.

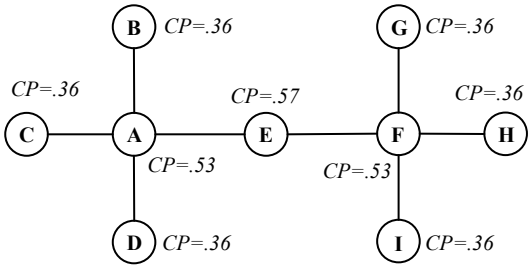


Figure 1.5 - Centralité de proximité (CP) pour les nœuds du graphe *G1*

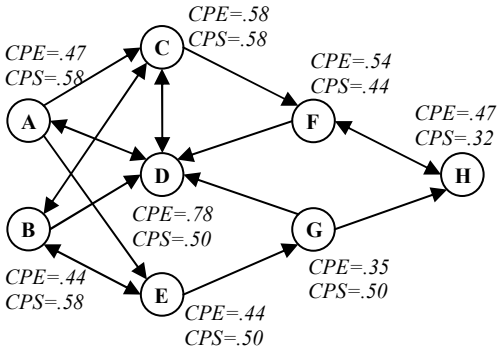


Figure 1.6 - Centralités de proximité entrante (CPE) et sortante (CPS) pour les nœuds du graphe *G2*

1.2.3 Centralité d'intermédiarité

La *centralité d'intermédiarité* [Freeman 79] est une autre mesure de centralité globale proposée par Freeman. L'intuition de cette mesure est que, dans un graphe, un nœud est d'autant plus important qu'il est nécessaire de le traverser pour aller d'un nœud quelconque à un autre. Plus précisément, un sommet ayant une forte centralité d'intermédiarité est un sommet par lequel passe un grand nombre de chemins géodésiques (i.e. chemins les plus courts) dans le graphe. Dans un réseau social, un acteur ayant une forte centralité d'intermédiarité est un sommet tel qu'un grand nombre d'interactions entre des sommets non adjacents dépend de lui [Borgatti and Everett 06]. Dans un réseau de communication, la centralité d'intermédiarité d'un nœud peut être considérée comme la probabilité qu'une information transmise entre deux nœuds passe par ce nœud intermédiaire [Borgatti and Everett 06].

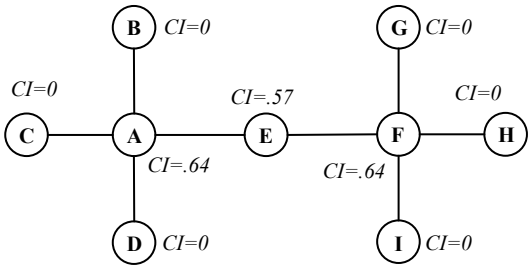


Figure 1.7 - Centralités d'intermédiarité (CI) pour les nœuds du graphe *G1*

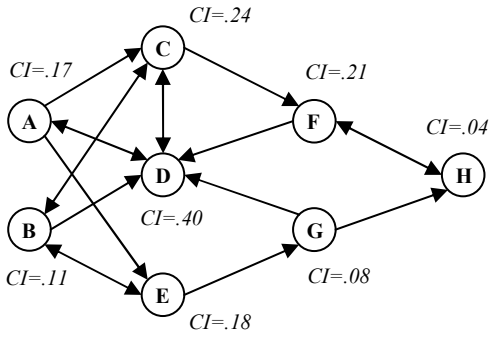


Figure 1.8 - Centralité d'intermédiarité (CI) pour les nœuds du graphe *G2*

Soit $G=(V,E)$ un graphe (orienté ou non) d'ordre *N*. La centralité d'intermédiarité d'un nœud $v_i \in V$ est définie par :

$$C^{\text{int}}(v_i)=\sum_{j=1}^{N}\sum_{k=1}^{N}\frac{g_{jk}(v_i)}{g_{jk}} \tag{1.3}$$

où $g_{jk}(v_i)$ est le nombre total de chemins géodésiques entre les nœuds v_j et v_k qui passent par le nœud v_i, et g_{jk} est le nombre total de chemins géodésiques entre les nœuds v_j et v_k.

Les figures 1.7 et 1.8 indiquent respectivement la centralité d'intermédiarité pour les nœuds des graphes jouets *G1* et *G2*. Nous remarquons que les nœuds A et F sont les plus importants dans le graphe *G1* ; ces deux nœuds sont en effet les plus traversés par les chemins géodésiques entre les nœuds du graphe. Pour le graphe *G2*, le nœud D est celui par lequel passe le plus grand nombre de chemins géodésiques entre les nœuds du graphe ; il possède par conséquent la plus forte centralité. Il est également intéressant de noter pour le graphe *G2*, que les nœuds E et F possèdent des centralités d'intermédiarité différentes bien qu'ils aient le même nombre de liens entrants et de liens sortants.

La centralité d'intermédiarité est basée sur l'hypothèse que les nœuds ne communiquent ou interagissent entre eux qu'à travers les chemins les plus courts. Certains chercheurs ont alors proposé de modifier cette hypothèse afin de prendre en compte le fait que les nœuds peuvent interagir en utilisant des chemins autres que les chemins géodésiques. Par exemple, Freeman [Freeman 91] a proposé la centralité du *flux d'intermédiarité* qui n'utilise pas que les chemins géodésiques mais plutôt tous les chemins indépendants entre deux nœuds i.e. les chemins dont les ensembles d'arcs sont disjoints.

1.2.4 Centralité spectrale

L'approche proposée par Bonacich [Bonacich 72][Bonacich 07] pour le calcul de centralité dans un graphe est très différente des autres approches que nous avons présentées. Il a en effet suggéré l'idée que la centralité d'un nœud soit déterminée par la centralité des nœuds auxquels il est connecté. Dans un réseau social, cela correspond à l'idée qu'un acteur

est d'autant plus important qu'il est connecté à des acteurs qui sont eux même importants. Il s'agit en fait d'une extension de la centralité du degré dans laquelle on ne donne pas le même poids aux nœuds voisins. Pour la mise en œuvre de ce principe, Bonacich propose de considérer la centralité d'un nœud comme étant dépendante de la combinaison linéaire des centralités de ses nœuds voisins.

Soit $G=(V,E)$ un graphe d'ordre *N* représenté par sa matrice d'adjacence $\mathbf{A}$. Dans le cas où le graphe *G* est non-orienté, la centralité spectrale $C^{spe}(v_i)$ d'un nœud $v_i \in V$ est donnée par l'équation :

$$\mu C^{spe}(v_i) = a_{1i}C^{spe}(v_1) + a_{2i}C^{spe}(v_2) + \cdots + a_{ni}C^{spe}(v_n) \tag{1.4}$$

où μ est un réel strictement positif.

Dans le cas où le graphe G est orienté, chaque nœud $v_i \in V$ possède alors deux mesures de centralité spectrale : une par rapport aux liens sortants et une par rapport aux liens entrants. Elles sont données respectivement par les équations suivantes :

$$\mu C_{out}^{spe}(v_i) = a_{i1}C_{out}^{spe}(v_1) + a_{i2}C_{out}^{spe}(v_2) + \cdots + a_{in}C_{out}^{spe}(v_n)$$

$$\lambda C_{in}^{spe}(v_i) = a_{1i}C_{in}^{spe}(v_1) + a_{2i}C_{in}^{spe}(v_2) + \cdots + a_{ni}C_{in}^{spe}(v_n)$$

où μ et λ sont des réels strictement positifs.

Le calcul de la centralité spectrale d'un nœud dans un graphe (orienté ou non) nécessite donc de résoudre un système d'équations qui peut être représenté en termes de matrices de la manière suivante :

$$\mu \mathbf{c}^{spe} = \mathbf{M}^T \mathbf{c}^{spe} \tag{1.5}$$

où μ est un réel strictement positif, $\mathbf{c}^{spe}$ est le vecteur de centralité spectrale ; $\mathbf{M}$ correspond soit à la matrice d'adjacence dans le cas du calcul de la centralité entrante, soit à la transposée de la matrice d'adjacence dans le cas du calcul de la centralité sortante (dans le cas où le graphe est non-orienté, nous avons $\mathbf{M} = \mathbf{M}^T$).

Pour résoudre l'équation 1.5, Bonacich montre que le vecteur de centralité spectrale $\mathbf{c}^{spe}$ correspond en fait au vecteur propre dominant (ou principal) de la matrice $\mathbf{M}$ (d'où les appellations de *centralité du vecteur propre* ou de *centralité spectrale*). Pour le calcul du vecteur $\mathbf{c}^{spe}$, il est possible d'utiliser par exemple la méthode des puissances itérées qui est particulièrement efficace lorsque la matrice $\mathbf{M}$ est creuse.

La centralité spectrale pour les nœuds des graphes jouets *G1* et *G2* est indiquée respectivement par les figures 1.9 et 1.10. Nous remarquons que les nœuds les plus importants dans le graphe *G1* sont les nœuds A et F. Pour le graphe *G2*, les nœuds A et B possèdent la plus forte centralité par rapport aux liens sortants, tandis que le nœud D possède la plus forte centralité par rapport aux liens entrants. Mais le plus intéressant avec ce graphe est que le nœud A est plus important que le nœud G par rapport aux liens entrants alors qu'ils ont tous

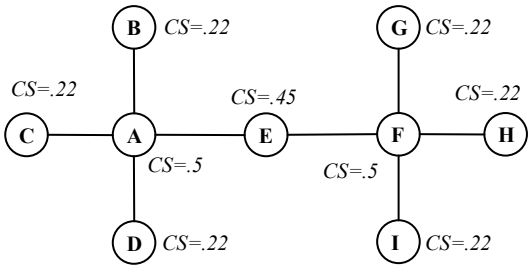


Figure 1.9 - Centralité spectrale (CS) pour les nœuds du graphe *G1*

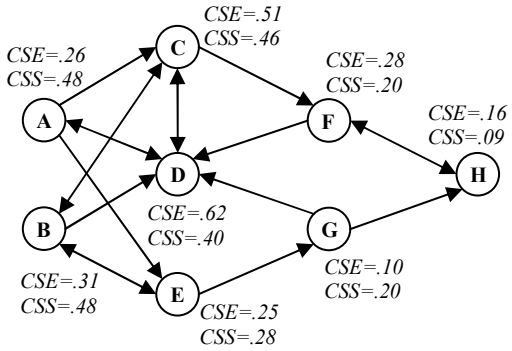


Figure 1.10 - Centralités spectrales entrante (CSE) et sortante (CSS) pour les nœuds du graphe *G2*

les deux le même nombre de liens entrants (égal à un). Cela s'explique par le fait que le nœud A est pointé par le nœud D qui est lui même plus important que le nœud E qui pointe vers G.

1.2.5 Limites des mesures de centralité issues de l'ARS

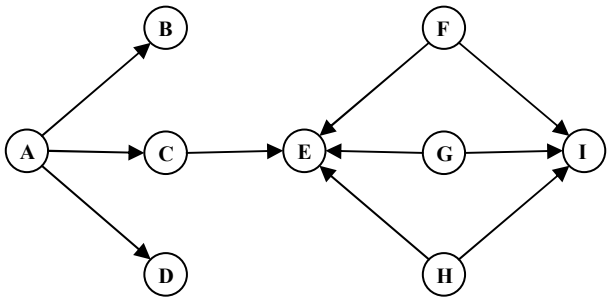
Afin d'illustrer les limites des quatre mesures de centralité que nous avons présentées, considérons le graphe jouet *G3* de la figure 1.11 où les différentes mesures de centralité pour chacun des nœuds de ce graphe sont indiquées.

Notre première remarque concerne la centralité de proximité qui n'est en fait utilisable que si le graphe est fortement connexe. Le graphe *G3* n'étant pas fortement connexe, il n'est par conséquent pas possible de calculer la centralité de proximité puisque les distances géodésiques entre certains nœuds sont indéfinies. Par exemple, la distance entre les nœuds E et I est indéfinie puisqu'il n'existe aucun chemin entre les deux nœuds.

Concernant la centralité d'intermédiarité, les résultats indiquent que tous les nœuds ont une centralité nulle à l'exception du nœud C. Cela est dû au fait que le nœud C est le seul à avoir à la fois des liens entrants et des liens sortants. D'après la définition de la centralité d'intermédiarité, il est évident que tout nœud qui ne possède pas au moins un lien entrant et un lien sortant aura une importance nulle.

Quant à la centralité spectrale, la figure 1.11 indique que tous les nœuds ont une importance nulle. Ce résultat inattendu est généralement obtenu lorsque le graphe est orienté et que certains nœuds ne possèdent qu'un seul type de liens (i.e. soit entrants soit sortants). Ces nœuds ne peuvent alors pas contribuer au calcul de la centralité. Par exemple, les nœuds A, F, G et H n'ont pas de liens entrants, ce qui fait qu'ils ont une centralité entrante nulle. Ces nœuds ne vont alors pas contribuer au calcul de la centralité spectrale des nœuds qu'ils pointent i.e. B, C, D, E et I. Ces derniers nœuds se retrouvent alors aussi avec une centralité nulle. Le même raisonnement peut être fait pour le calcul de la centralité sortante. Notons au passage que les valeurs et vecteurs propres d'une matrice asymétrique peuvent être complexes contrairement aux matrices symétriques dont les valeurs et vecteurs propres sont toujours réels [Golub and Van Loan 96].

Finalement, parmi les quatre mesures de centralité que nous avons présentées jusque-là, il n'y aurait que la centralité du degré qui soit adaptée aux graphes de documents car ces derniers sont orientés et généralement non connexes (au meilleur des cas, ils peuvent être faiblement connexes). De plus, dans un graphe de documents, les sommets ne possèdent pas tous des liens entrants et des liens sortants. A titre d'exemple, un article scientifique ou une page web nouvellement créés ne vont pas avoir de liens entrants au moment de leur publication. Cependant, à la fin des années 90, un grand nombre de chercheurs se sont penchés sur le problème du calcul de centralité dans les graphes de documents afin de proposer d'autres mesures. Comme nous allons le voir dans la section suivante, la plupart de ces mesures (pour ne pas dire toutes) sont des variantes de la centralité spectrale.



Nœud	*A*	*B*	*C*	*D*	*E*	*F*	*G*	*H*	*I*
Centralité de degré sortant	0.38	0	0.13	0	0	0.26	0.26	0.26	0
Centralité du degré entrant	0	0.13	0.13	0.13	0.5	0	0	0	0.38
Centralité d'intermédiarité	0	0	0.02	0	0	0	0	0	0
Centralité spectrale sortante	0	0	0	0	0	0	0	0	0
Centralité spectrale entrante	0	0	0	0	0	0	0	0	0

Figure 1.11 - Centralités de degré, d'intermédiarité et spectrale pour les nœuds du graphe G3

1.3 Mesures de centralité issues de la Recherche d'Information (RI)

A la fin des années 90, plusieurs chercheurs ont envisagé d'utiliser les liens entre documents (en plus des contenus textuels) afin d'améliorer les performances des moteurs de recherche sur le web. La pertinence d'un document par rapport à une requête est alors calculée en combinant la similarité du document par rapport à la requête avec la centralité du document [Baeza-Yates and Ribeiro-Neto 99][Chakrabarti 02][Manning et al. 08]. Nous présentons ci-dessous quelques uns des algorithmes les plus connus pour le calcul de centralité dans les graphes de documents. Ces algorithmes sont tous basés sur l'idée que l'importance d'un document est relative à l'importance des documents auxquels il est connecté.

1.3.1 PageRank

Proposé à la fin des années 90 par les deux informaticiens Brin et Page, PageRank [Brin and Page 98] est l'un des algorithmes d'analyse de liens qui a le plus marqué le domaine de la recherche d'information sur le web. Il a d'ailleurs été classé parmi le top 10 des algorithmes de data mining lors de la conférence ICDM 2006 [Wu et al. 08]. C'est notamment grâce à cet algorithme que le moteur de recherche de Google a connu le succès qu'il a aujourd'hui.

Dans sa version simplifiée, l'algorithme PageRank considère que l'importance (appelée aussi popularité ou encore PageRank) d'une page est fonction des popularités des pages qui la pointent (ou la citent). Plus précisément, le PageRank simplifié d'une page p_i est donné par [Zhang et al. 05] :

$$PR_S\left(p_i\right) = \sum_{p_j \in in(p_i)} \frac{PR_S\left(p_j\right)}{d^{out}\left(p_j\right)} \tag{1.6}$$

où $in\left(p_i\right)$ représente l'ensemble des pages qui pointent vers la page p_i et $d^{out}\left(p_j\right)$ représente le degré sortant de la page p_j.

L'algorithme 1.1 indique les différentes étapes de calcul du PageRank simplifié. Les étapes 1 à 2 de l'algorithme permettent de transformer la matrice d'adjacence $\mathbf{A}$ en une matrice stochastique $\mathbf{A}_l$. Cette transformation est réalisée en normalisant les lignes de la matrice d'adjacence $\mathbf{A}$ (étape 5) tel que :

$$\left(a_l\right)_{ij} = \frac{a_{ij}}{\sum_{k=1}^{N} a_{ik}} \tag{1.7}$$

Le but de cette normalisation est d'atténuer l'effet des pages ayant un grand nombre de liens sortants. En d'autres termes, cela revient à considérer que chaque page possède un seul "vote" qui sera distribué de manière égale aux pages pointées.

Les étapes 3 à 7 de l'algorithme correspondent à l'application de la méthode des puissances pour le calcul du vecteur propre dominant de la matrice $\mathbf{A}_l^T$. Dans la dernière étape, le vecteur du PageRank $\mathbf{p}$ est normalisé afin que la somme des PageRank de toutes les pages soit égale à 1.

La figure 1.12 montre les résultats obtenus en appliquant l'algorithme du PageRank simplifié avec le graphe jouet *G2*. La figure indique que le nœud D est le plus populaire (ou le plus important) car il est pointé par plusieurs nœuds qui sont eux même importants. Nous remarquons aussi que le nœud A possède un PageRank plus important que les nœuds B, F et H bien que ces derniers aient plus de liens entrants que A. Cela s'explique par le fait que A est pointé par un nœud (à savoir D) qui est plus important que les nœuds qui pointent vers B (C et E), E (A et B) ou H (F et G).

Algorithme 1.1 : Algorithme du PageRank simplifié

Entrée : - une matrice d'adjacence irréductible et apériodique $\mathbf{A} \in \mathbb{R}^{N \times N}$
Sortie : vecteur du PageRank simplifié $\mathbf{p}$

début

1. $\mathbf{D}_l \leftarrow diag\left(\mathbf{A} \times \mathbf{1}_{N\times 1}\right)$ // $\mathbf{1}_{N\times 1}$ est un vecteur colonne de dimension N contenant des uns
2. $\mathbf{A}_l \leftarrow \mathbf{D}_l^{-1}\mathbf{A}$
3. $\mathbf{p}^{(0)} \leftarrow \frac{1}{N}\mathbf{1}_{N\times 1}$, $t \leftarrow 1$
4. **répéter**
5. $\quad \mathbf{p}^{(t)} \leftarrow \mathbf{A}_l^{\,T}\mathbf{p}^{(t-1)}$
6. $\quad t \leftarrow t+1$
7. **jusqu'à** *convergence*
8. $\mathbf{p}^{(t)} \leftarrow \frac{\mathbf{p}^{(t)}}{\left\|\mathbf{p}^{(t)}\right\|_1}$ // $\left\|\mathbf{x}\right\|_1 = \sum_{i=1} x_i$ est la norme L1 du vecteur $\mathbf{x}$

fin

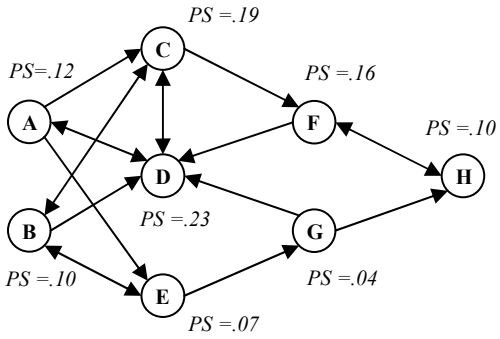


Figure 1.12 – PageRank simplifié (PS) pour les nœuds du graphe *G2*

Une interprétation classique de l'algorithme PageRank est celle du surfeur aléatoire [Langville and Meyer 05]. Un surfeur aléatoire représente un utilisateur virtuel qui navigue à travers le graphe de liens en suivant à chaque fois un des hyperliens de la page courante. Dans la version simplifiée du PageRank, le surfeur aléatoire choisit à chaque fois de suivre de manière équiprobable un lien parmi les différents liens sortants de la page sur laquelle il se trouve.

En utilisant la métaphore du surfeur aléatoire, l'importance d'une page est alors déterminée par la proportion de temps qu'aura passé le surfeur aléatoire sur cette page à l'instant t lorsque $t \to \infty$. En d'autres termes, une page est considérée comme étant importante si elle est fréquemment visitée par le surfeur aléatoire. Formellement, ce principe appelé aussi "marche aléatoire" est modélisé par une chaîne de Markov [Baldi et al. 03]. Une chaîne de Markov décrit un processus stochastique à temps discret. Elle est définie par un ensemble d'états ainsi qu'une matrice de transition indiquant la probabilité de passage d'un état à un autre. Dans [Brin and Page 98], Brin et Page montrent que le vecteur du PageRank

correspond à la distribution stationnaire d'une chaîne de Markov construite à partir du graphe de documents. Rappelons que la *distribution stationnaire* d'une chaîne de Markov définie par sa matrice de transitions $\mathbf{P}$, est un vecteur $\boldsymbol{\pi}$ tel que :

$$\boldsymbol{\pi} = \mathbf{P}^T \boldsymbol{\pi} \tag{1.8}$$

Une entrée π_i du vecteur $\boldsymbol{\pi}$ indique la probabilité que la chaîne soit à l'état *i* à n'importe quel instant. L'existence d'une distribution stationnaire unique n'est cependant garantie que si la chaîne de Markov est irréductible et apériodique. C'est pourquoi la version simplifiée du PageRank nécessite que la matrice d'adjacence ait ces deux propriétés. Dans le cas où ces deux conditions ne sont pas vérifiées, le PageRank simplifié donne (en général) des résultats contre-intuitifs comme le montre la figure 1.13. Les résultats indiqués par cette figure s'expliquent en partie par le fait que plusieurs nœuds du graphe ne possèdent pas de liens sortants. Ces pages, appelées aussi pages puits, posent problème au surfeur aléatoire car une fois qu'il les visite, il ne pourra plus les quitter. Une autre raison expliquant ce résultat est que si par exemple le surfeur aléatoire commence sa marche au nœud A, il ne pourra jamais atteindre certains nœuds du graphe tels que F et I.

Cependant, les deux conditions d'irréductibilité et d'apériodicité n'étant que très rarement satisfaites en pratique, Brin et Page ont alors proposé une variante du PageRank simplifié (que nous appelons PageRank pratique) qui consiste à modifier le graphe initial afin de le rendre irréductible et apériodique. Le PageRank pratique est en fait basé sur un nouveau modèle de marche aléatoire dans lequel le surfeur aléatoire choisit à chaque étape :

- soit de suivre un des liens sortants de la page courante avec une probabilité égale à α.
- soit de visiter une page quelconque du graphe avec une probabilité égale à $(1-\alpha)$.

où α, appelé paramètre d'amortissement ou facteur de zap, est un réel compris entre 0 et 1. Lorsque $\alpha = 1$, nous remarquons que ce modèle est équivalent à celui du PageRank simplifié.

L'algorithme 1.2 décrit le processus itératif permettant de calculer le vecteur du PageRank pratique. L'étape 3 de l'algorithme modifie la matrice d'adjacence en rajoutant des liens aux nœuds ayant un degré sortant nul. A l'étape 5, l'algorithme construit la matrice irréductible et apériodique G, appelée parfois matrice de Google [Langville and Meyer 06], en rajoutant des liens (pondérés) entre tous les nœuds du graphe. La suite de l'algorithme consiste à appliquer l'algorithme du PageRank simplifié avec la matrice G en entrée.

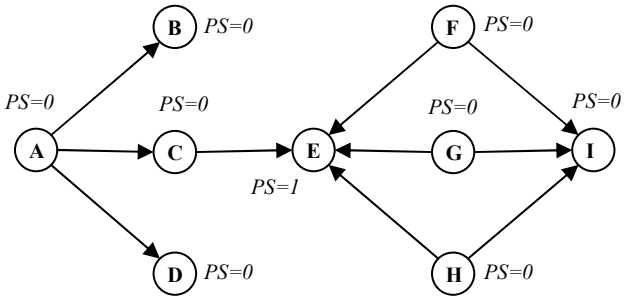


Figure 1.13 – PageRank simplifié (PS) pour les nœuds du graphe G3

Algorithme 1.2 : Algorithme du PageRank pratique

Entrée : - une matrice d'adjacence $\mathbf{A} \in \mathbb{R}^{N \times N}$
- paramètre d'amortissement $\alpha \in]0,1[$

Sortie : vecteur du PageRank pratique $\mathbf{p}$

début

1. $\mathbf{D}_l \leftarrow diag\left(\mathbf{A} \times \mathbf{1}_{N\times 1}\right)$ // $\mathbf{1}_{N\times 1}$ est un vecteur colonne de dimension N contenant des uns
2. **pour** $i = 1, \ldots, N$ **faire**
3. $\quad$ **si** $\left(d_l\right)_{ii} = 0$ **alors** $\mathbf{a}_{i.} \leftarrow \frac{1}{N}\mathbf{1}_{1\times N}$, $\left(d_l\right)_{ii} \leftarrow 1$ // $\mathbf{a}_{i.}$ est la ligne i de la matrice $\mathbf{A}$
4. **fin**
5. $\mathbf{G} \leftarrow \alpha \mathbf{D}_l^{-1}\mathbf{A} + \left(1-\alpha\right)\frac{1}{N}\mathbf{1}_{N\times N}$
6. $\mathbf{p}^{(0)} \leftarrow \frac{1}{N}\mathbf{1}_{N\times 1}$, $t \leftarrow 1$
7. **répéter**
8. $\quad \mathbf{p}^{(t)} \leftarrow \mathbf{G}^T\mathbf{p}^{(t-1)}$
9. $\quad t \leftarrow t+1$
10. **jusqu'à** *convergence*
11. $\mathbf{p}^{(t)} \leftarrow \frac{\mathbf{p}^{(t)}}{\left\|\mathbf{p}^{(t)}\right\|_1}$ // $\left\|\mathbf{x}\right\|_1 = \sum_{i=1} x_i$ est la norme L1 du vecteur $\mathbf{x}$

fin

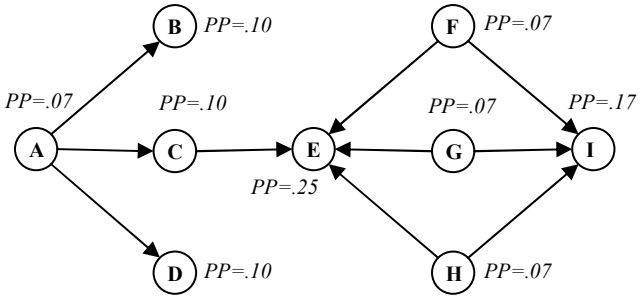


Figure 1.14 – PageRank pratique (PP) pour les nœuds du graphe G3

La modification introduite dans l'algorithme du PageRank pratique va ainsi permettre au surfeur aléatoire de quitter les pages puits et aussi de visiter les pages qui n'ont pas de liens entrants. Concernant le facteur d'amortissement α, Brin et Page suggèrent d'utiliser la valeur 0.85 en arguant que cette valeur offre un bon compromis entre la précision des résultats et la vitesse de convergence de l'algorithme.

La figure 1.14 indique les résultats obtenus en calculant le PageRank pratique pour les nœuds du graphe jouet *G3* (pour le graphe jouet *G2*, le PageRank pratique donne des résultats similaires à ceux de l'algorithme du PageRank simplifié). Nous remarquons que le nœud E est

le plus important car il est pointé par plusieurs nœuds ayant un degré d'importance non nul. Il est par ailleurs intéressant de noter ici qu'avec le PageRank pratique, tous les nœuds auront un PageRank différent de zéro y compris ceux qui ne possèdent pas de liens entrants.

Bien qu'initialement proposé pour l'analyse de liens hypertextes, l'algorithme PageRank a été utilisé plus récemment pour l'analyse d'autres types de graphes notamment pour l'analyse des références bibliographiques [Fiala et al. 08][Ma et al. 08].

Une critique que l'on peut faire à l'algorithme PageRank est la modification considérable apportée au graphe initial, notamment en rajoutant de manière équiprobable des liens entre tous les documents. D'autre part, le paramètre d'amortissement α peut parfois avoir une grande influence sur les résultats obtenus comme le montrent plusieurs travaux ([Chen et al. 07] et [Maslov and Redner 08]), ce qui pose le problème du choix de la valeur de ce paramètre.

Enfin, nous noterons que l'algorithme PageRank n'est en réalité qu'une variante de la centralité spectrale de Bonacich, variante qui commence d'abord par modifier la matrice d'adjacence initiale avant de calculer son vecteur propre dominant.

1.3.2 HITS (Hypertext Induced Topic Search)

Dans le cadre du projet Clever [Kumar et al. 06] développé par IBM en 1998, Kleinberg [Kleinberg 98][Kleinberg 99a] a proposé l'algorithme HITS dont l'idée est d'exploiter la structure du web afin d'améliorer la qualité de la recherche d'information. Cependant, à la différence de PageRank qui assigne à chaque page un seul degré d'importance, l'algorithme HITS caractérise chaque page par deux degrés d'importance. Ces deux degrés, que Kleinberg appelle degrés d'*autorité* et d'*hubité*, sont respectivement des mesures de centralité par rapport aux liens entrants et aux liens sortants.

L'algorithme HITS considère que le degré d'autorité d'une page est égal à la somme des degrés d'hubité des pages qui la pointent (ou la citent). En d'autres termes, une page est une bonne autorité si elle est pointée par de bons hubs. Plus précisément, le degré d'autorité d'une page p_i est défini par :

$$A(p_i) = \sum_{p_j \in in(p_i)} H(p_j) \tag{1.9}$$

où $in(p_i)$ représente l'ensemble des pages qui pointent vers la page p_i et $H(p_j)$ représente le degré d'hubité de la page p_j.

De manière similaire, HITS considère que le degré d'hubité d'une page est égal à la somme des degrés d'autorité des pages qu'elle pointe. Cela sous-entend qu'une page est un bon hub si elle pointe vers de bonnes autorités. Ainsi, le degré d'hubité d'une page p_i est défini par :

$$H(p_i) = \sum_{p_j \in out(p_i)} A(p_j) \tag{1.10}$$

où $out(p_i)$ représente l'ensemble des pages pointées par la page p_i et $A(p_j)$ représente le degré d'autorité de la page p_j.

Algorithme 1.3 : L'algorithme HITS

Entrée : - une matrice d'adjacence $\mathbf{A} \in \mathbb{R}^{N \times N}$
Sortie : vecteurs d'autorité **o** et d'hubité **h**
début
1. $\mathbf{o}^{(0)} \leftarrow \mathbf{1}_{N\times 1}$, $\mathbf{h}^{(0)} \leftarrow \mathbf{1}_{N\times 1}$, $t \leftarrow 1$
2. **répéter**
3. $\quad \mathbf{o}^{(t)} \leftarrow \mathbf{A}^T \mathbf{h}^{(t-1)}$
4. $\quad \mathbf{h}^{(t)} \leftarrow \mathbf{A}\mathbf{o}^{(t-1)}$
5. $\quad \mathbf{o}^{(t)} \leftarrow \frac{\mathbf{o}^{(t)}}{\left\|\mathbf{o}^{(t)}\right\|_2}$, $\mathbf{h}^{(t)} \leftarrow \frac{\mathbf{h}^{(t)}}{\left\|\mathbf{h}^{(t)}\right\|_2}$ // $\left\|\mathbf{x}\right\|_2 = \sqrt{\sum_{i=1} x_i^2}$ est la norme L2 du vecteur **x**
6. $\quad t \leftarrow t+1$
7. **jusqu'à** *convergence*

fin

L'algorithme 1.3 présente les différentes étapes de calcul des vecteurs d'autorité et d'hubité par HITS. L'algorithme part d'un vecteur d'autorité initial, puis effectue une itération entre une mise à jour du vecteur **h** (vecteur des hubs) en utilisant le vecteur **o** (vecteur des autorités) et une mise à jour du vecteur **o** en utilisant le vecteur **h**. Dans son article de référence, Kleinberg [Kleinberg 99] parle de principe de *renforcement mutuel* entre les hubs et les autorités pour décrire ce processus itératif de mise à jour des vecteurs **o** et **h**.

Kleinberg montre notamment que le vecteur d'autorité **o** calculé par son algorithme correspond au vecteur propre principal de la matrice $\mathbf{O}_{HITS} = \mathbf{A}^T\mathbf{A}$ (que nous appelons matrice des autorités de HITS), où **A** est la matrice d'adjacence. Il suffit en fait de remplacer le vecteur **h** dans la ligne 3 par la formule de la ligne 4 pour s'apercevoir que HITS n'est rien d'autre qu'une application de la méthode des puissances pour le calcul du vecteur propre principal de la matrice $\mathbf{O}_{HITS}$. Le même raisonnement peut être utilisé pour montrer que le vecteur des hubs **h** calculé par HITS correspond au vecteur propre principal de la matrice $\mathbf{H}_{HITS} = \mathbf{A}\mathbf{A}^T$ (que nous appelons matrice des hubs de HITS).

La matrice $\mathbf{O}_{HITS}$ (resp. $\mathbf{H}_{HITS}$) est très proche de la matrice de co-citation [Small 73] (resp. de couplage bibliographique [Kessler 63]) utilisée en bibliométrie. Ding et al. [Ding et al. 02] montrent en effet que :

$$\mathbf{O}_{HITS} = \mathbf{M}_{coc} + \mathbf{D}_{in} \quad , \quad \mathbf{H}_{HITS} = \mathbf{M}_{bib} + \mathbf{D}_{out}$$

où :

- $\mathbf{M}_{coc}$ est la matrice de co-citation. Une entrée (i, j) de cette matrice indique le nombre de fois que les documents *i* et *j* sont co-cités par d'autres documents.
- $\mathbf{M}_{bib}$ est la matrice de couplage bibliographique. Une entrée (i, j) de cette matrice indique le nombre de fois que les documents *i* et *j* citent le même document.
- $\mathbf{D}_{in}$ est une matrice diagonale dans laquelle une entrée (i,i) indique le nombre de liens entrants du document *i*.

- $\mathbf{D}_{out}$ est une matrice diagonale dans laquelle une entrée (i,i) indique le nombre de liens sortants du document *i*.

La figure 1.15 indique les degrés d'autorité et d'hubité calculés par HITS pour les nœuds du graphe jouet *G2*. Concernant les degrés d'autorité, nous remarquons que le nœud D est le plus important car il est pointé par plusieurs nœuds qui ont un fort degré d'hubité. Nous remarquons aussi que le nœud E est plus important que le nœud H bien qu'ils aient le même nombre de liens entrants. Cela est dû au fait que les nœuds qui pointent vers E, à savoir A et B, ont un degré d'hubité supérieur à celui des nœuds F et G qui pointent vers H. Par rapport aux degrés d'hubité, la figure 1.15 montre que A et B sont les meilleurs hubs : ils pointent en effet vers plusieurs nœuds ayant un fort degré d'autorité. Notons enfin que les nœuds A et B ont le même degré d'hubité car ils pointent vers les mêmes nœuds à savoir C, D et E.

Nous reportons sur la figure 1.16 les résultats obtenus en analysant le graphe jouet *G3* en utilisant l'algorithme HITS. Nous remarquons que certains résultats sont étonnants car le nœud A, par exemple, a un degré d'hubité nul alors qu'il possède trois liens sortants. De même, les nœuds B et D ont tous les deux un lien entrant et malgré cela l'algorithme HITS leur assigne un degré d'autorité nul. Une telle situation où certains nœuds obtiennent un degré d'autorité ou d'hubité faible (voire nul comme c'est le cas ici) alors qu'ils devraient avoir une importance non négligeable a été mise en évidence par Lempel et Moran [Lempel and Moran 00]. Ces derniers appellent ce "défaut" de HITS : problème de l'effet TKC ("Tightly Knit Community" ou communauté fortement connectée).

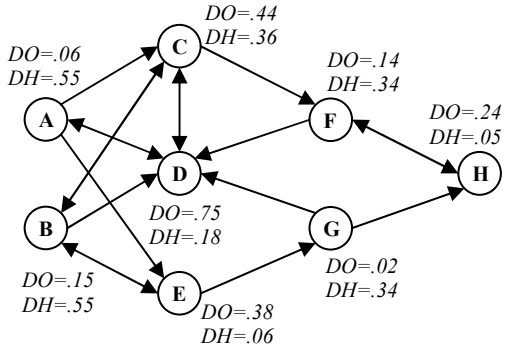


Figure 1.15 – Degrés d'autorité (DO) et d'hubité (DH) calculés par HITS pour les nœuds du graphe *G2*

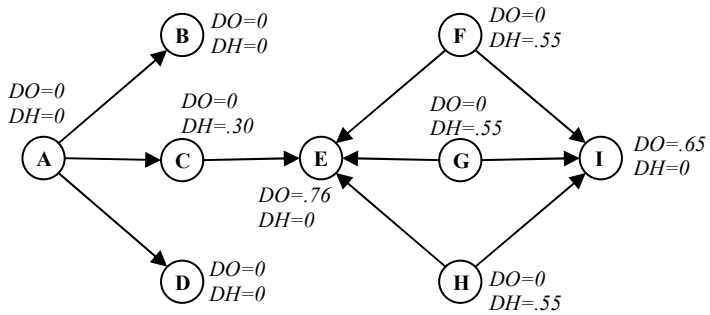


Figure 1.16- Degrés d'autorité (DO) et d'hubité (DH) calculés par HITS pour les nœuds du graphe *G3*

Un autre problème lié à l'algorithme HITS concerne sa convergence. Cet aspect est étudié en détail dans [Farahat et al. 05] où les auteurs montrent à travers plusieurs exemples que la convergence de l'algorithme peut poser problème avec certains graphes. Farahat et al. prouvent de plus, en se basant sur le théorème de Perron-Frobenius, que l'algorithme HITS converge vers un vecteur d'autorité **o** (resp. d'hubité **h**) unique si et seulement si la matrice $\mathbf{O}_{HITS}$ (resp. $\mathbf{H}_{HITS}$) est irréductible.

Par ailleurs, mentionnons qu'il est possible d'interpréter le calcul du vecteur d'autorité **o** (resp. d'hubité **h**) effectué par HITS de la façon suivante : dans un premier temps, l'algorithme de Kleinberg construit une matrice symétrique, en l'occurrence $\mathbf{O}_{HITS}$ (resp. $\mathbf{H}_{HITS}$), qui préserve l'information sur les liens entrants (resp. sortants) contenue dans la matrice d'adjacence. Dans un second temps, l'algorithme calcule la centralité spectrale de Bonacich pour les nœuds du graphe représenté par la matrice $\mathbf{O}_{HITS}$ (resp. $\mathbf{H}_{HITS}$).

1.3.3 SALSA (Stochastic Approach for Link Structure Analysis)

En s'inspirant des algorithmes HITS et PageRank, Lempel et Moran proposent dans [Lempel and Moran 00] l'algorithme SALSA. Celui-ci est similaire à HITS dans le sens où il calcule pour chaque nœud un degré d'autorité et un degré d'hubité, contrairement à PageRank qui ne calcule qu'une seule mesure d'importance. Quant à la similitude avec PageRank, elle réside dans le fait que SALSA soit basé sur le principe de la marche aléatoire pour le calcul des degrés d'autorité et d'hubité.

La principale motivation de l'algorithme SALSA est de résoudre le problème de la TKC dont souffre l'algorithme HITS. Ce problème, rappelons-le, se manifeste par une attribution "non équitable" des degrés d'importance ; certains nœuds qui sont supposés être importants se voient assigner un degré d'importance faible (voire nul).

Le calcul des degrés d'autorité (resp. d'hubité) par SALSA se fait en deux étapes. Dans un premier temps, l'algorithme construit une chaîne de Markov M_o (resp. M_h) à partir de la matrice d'adjacence **A** d'un graphe G (d'ordre N et de taille M). Dans un second temps, l'algorithme calcule la distribution stationnaire de cette chaîne de Markov. Les matrices de transition $\mathbf{O}_{SALSA}$ et $\mathbf{H}_{SALSA}$ des chaînes de Markov M_o et M_h respectivement sont définies par :

$$\mathbf{O}_{SALSA} = \mathbf{A}_c^{\,T}\mathbf{A}_l \quad , \quad \mathbf{H}_{SALSA} = \mathbf{A}_l\mathbf{A}_c^{\,T}$$

où $\mathbf{A}_c$ est la matrice obtenue en normalisant (avec la norme L1) les colonnes non nulles de la matrice $\mathbf{A}$; $\mathbf{A}_l$ est la matrice obtenue en normalisant (avec la norme L1) les lignes non nulles de la matrice $\mathbf{A}$.

Sachant que la distribution stationnaire **o** (resp. **h**) de la chaîne de Markov M_o (resp. M_h) correspond au vecteur propre principal de la matrice $\mathbf{O}_{SALSA}$ (resp. $\mathbf{H}_{SALSA}$), il est par conséquent possible de calculer le vecteur **o** (resp. **h**) en utilisant la méthode des puissances (sous réserve que les matrices $\mathbf{O}_{SALSA}$ et $\mathbf{H}_{SALSA}$ soient irréductibles).

Lempel et Moran montrent que si la chaîne de Markov M_o est irréductible, elle possède alors une distribution stationnaire unique $\mathbf{o} = (o_1, \ldots, o_N)$ qui vérifie pour tout $i \in V$:

$$o_i = \frac{d^{in}(i)}{M} \tag{1.11}$$

où $d^{in}(i)$ représente le degré entrant du nœud *i*.

Ils montrent par ailleurs que si la chaîne de Markov M_h est irréductible, elle possède alors une distribution stationnaire unique $\mathbf{h} = (h_1, \ldots, h_N)$ qui vérifie pour tout $i \in V$:

$$h_i = \frac{d^{out}(i)}{M} \tag{1.12}$$

où $d^{out}(i)$ représente le degré sortant du nœud *i*.

Dans le cas où la chaîne de Markov M_o (resp. M_h) n'est pas irréductible, c'est-à-dire que son graphe contient plusieurs composantes connexes, Lempel et Moran proposent une méthode simple permettant de calculer le vecteur d'autorité **o** (resp. d'hubité **h**). La méthode consiste à appliquer l'algorithme SALSA sur chacune des composantes connexes puis à multiplier le degré d'importance de chaque nœud par un facteur proportionnel à la taille de la composante contenant ce nœud. Plus précisément, si le graphe de la chaîne de Markov M_o contient *l* composantes connexes $C_1, C_2 \ldots, C_l$, le degré d'autorité o_i d'un nœud $i \in C_k$ est donné par :

$$o_i = \frac{|C_k|}{N} \frac{d^{in}(i)}{d^{in}(C_k)} \tag{1.13}$$

De même, si le graphe de la chaîne de Markov M_h contient *m* composantes connexes $P_1, P_2 \ldots, P_m$, le degré d'hubité h_i d'un nœud $i \in P_k$ est donné par :

$$h_i = \frac{|P_k|}{N} \frac{d^{out}(i)}{d^{out}(P_k)} \tag{1.14}$$

Dans leur manière de présenter l'algorithme SALSA, Lempel et Moran sous-entendent que le calcul du vecteur d'autorité est indépendant du vecteur d'hubité (et vice versa). Ils affirment d'ailleurs que c'est grâce à cette séparation entre le calcul des autorités et le calcul des hubs que l'algorithme SALSA parvient à éviter l'effet de la TKC. En réalité, nous pouvons montrer qu'il y a toujours un renforcement mutuel entre les hubs et les autorités ; il suffit pour cela de remarquer que l'algorithme SALSA est équivalent à une version de HITS où :

- le degré d'autorité d'une page est égal à la moyenne des degrés d'hubité des pages qui la pointent. En d'autres termes, une page est une bonne autorité si elle est pointée par de bons hubs *uniquement*. Cela revient à pénaliser les autorités qui sont pointées par de mauvais hubs. Plus précisément, le degré d'autorité d'une page p_i est défini par :

$$A(p_i) = \frac{1}{|in(p_i)|} \sum_{p_j \in in(p_i)} H(p_j) \tag{1.15}$$

où $in(p_i)$ représente l'ensemble des pages qui pointent vers la page p_i et $H(p_j)$ représente le degré d'hubité de la page p_j.

- le degré d'hubité d'une page est égal à la moyenne des degrés d'autorité des pages qu'elle pointe. En d'autres termes, une page est un bon hub si elle pointe vers de bonnes autorités *uniquement*. Cela revient à pénaliser les hubs qui pointent vers de mauvaises autorités. Plus précisément, le degré d'hubité d'une page p_i est défini par :

$$H(p_i) = \frac{1}{|out(p_i)|} \sum_{p_j \in out(p_i)} A(p_j) \tag{1.16}$$

où $out(p_i)$ représente l'ensemble des pages pointées par la page p_i et $A(p_j)$ représente le degré d'autorité de la page p_j.

En utilisant ces deux définitions des degrés d'autorité et d'hubité, nous obtenons alors l'algorithme 1.4 qui permet de calculer les vecteurs d'autorité et d'hubité. En remplaçant le vecteur **h** dans la ligne 9 par la formule de ligne 10, nous remarquons que le vecteur autorité **o** calculé par cet algorithme correspond au vecteur propre principal de la matrice $\mathbf{O}_{SALSA} = \mathbf{A}_c^{\,T}\mathbf{A}_l$. De même, nous remarquons que le vecteur **h** calculé par cet algorithme correspond au vecteur propre principal de la matrice $\mathbf{H}_{SALSA} = \mathbf{A}_l\mathbf{A}_c^{\,T}$. Cet algorithme nécessite toutefois que les matrices $\mathbf{O}_{SALSA}$ et $\mathbf{H}_{SALSA}$ soient irréductibles. Dans le cas où elles ne le sont pas, il faudra alors appliquer l'algorithme sur chaque composante connexe puis combiner les résultats des différentes composantes (méthode de Lempel et Moran).

La figure 1.17 indique les degrés d'autorité et d'hubité obtenus en appliquant l'algorithme SALSA avec le graphe jouet *G3*. Pour ce graphe, les matrice de transition $\mathbf{O}_{SALSA}$ et $\mathbf{H}_{SALSA}$ sont données par :

$$\mathbf{O}_{SALSA} = \begin{array}{c} \\ B \\ C \\ D \\ E \\ I \end{array} \begin{array}{c} \begin{array}{ccccc} B & C & D & E & I \end{array} \\ \begin{bmatrix} 0.33 & 0.33 & 0.33 & 0 & 0 \\ 0.33 & 0.33 & 0.33 & 0 & 0 \\ 0.33 & 0.33 & 0.33 & 0 & 0 \\ 0 & 0 & 0 & 0.62 & 0.37 \\ 0 & 0 & 0 & 0.50 & 0.50 \end{bmatrix} \end{array}$$

$$\mathbf{H}_{SALSA} = \begin{array}{c} \\ A \\ C \\ F \\ G \\ H \end{array} \begin{array}{c} \begin{array}{ccccc} A & C & F & G & H \end{array} \\ \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0.25 & 0.25 & 0.25 & 0.25 \\ 0 & 0.13 & 0.29 & 0.29 & 0.29 \\ 0 & 0.13 & 0.29 & 0.29 & 0.29 \\ 0 & 0.13 & 0.29 & 0.29 & 0.29 \end{bmatrix} \end{array}$$

Algorithme 1.4 : L'algorithme SALSA

Entrée : - une matrice d'adjacence $\mathbf{A} \in \mathbb{R}^{N \times N}$

Sortie : vecteurs d'autorité $\mathbf{o}$ et d'hubité $\mathbf{h}$

début

1. $\mathbf{D}_l \leftarrow diag\left(\mathbf{A} \times \mathbf{1}_{N\times 1}\right)$, $\mathbf{D}_c \leftarrow diag\left(\mathbf{A}^T \times \mathbf{1}_{N\times 1}\right)$
2. **pour** $i = 1, \ldots, N$ **faire**
3. | **si** $\left(d_l\right)_{ii} = 0$ **alors** $\left(d_l\right)_{ii} \leftarrow 1$ // Permet de rendre la matrice $\mathbf{D}_l$ inversible
4. | **si** $\left(d_c\right)_{ii} = 0$ **alors** $\left(d_c\right)_{ii} \leftarrow 1$ // Permet de rendre la matrice $\mathbf{D}_c$ inversible
5. **fin**
6. $\mathbf{A}_l \leftarrow \mathbf{D}_l^{-1}\mathbf{A}$, $\mathbf{A}_c \leftarrow \mathbf{A}\mathbf{D}_c^{-1}$
7. $\mathbf{o}^{(0)} \leftarrow \mathbf{1}_{N\times 1}$, $\mathbf{h}^{(0)} \leftarrow \mathbf{1}_{N\times 1}$, $t \leftarrow 1$
8. **répéter**
9. | $\mathbf{o}^{(t)} \leftarrow \mathbf{A}_c{}^T\mathbf{h}^{(t-1)}$
10. | $\mathbf{h}^{(t)} \leftarrow \mathbf{A}_l\mathbf{o}^{(t-1)}$
11. | $\mathbf{o}^{(t)} \leftarrow \dfrac{\mathbf{o}^{(t)}}{\left\|\mathbf{o}^{(t)}\right\|_2}$, $\mathbf{h}^{(t)} \leftarrow \dfrac{\mathbf{h}^{(t)}}{\left\|\mathbf{h}^{(t)}\right\|_2}$
12. | $t \leftarrow t+1$
13. **jusqu'à** *convergence*
14. $\mathbf{o}^{(t)} \leftarrow \dfrac{\mathbf{o}^{(t)}}{\left\|\mathbf{o}^{(t)}\right\|_1}$, $\mathbf{h}^{(t)} \leftarrow \dfrac{\mathbf{h}^{(t)}}{\left\|\mathbf{h}^{(t)}\right\|_1}$

fin

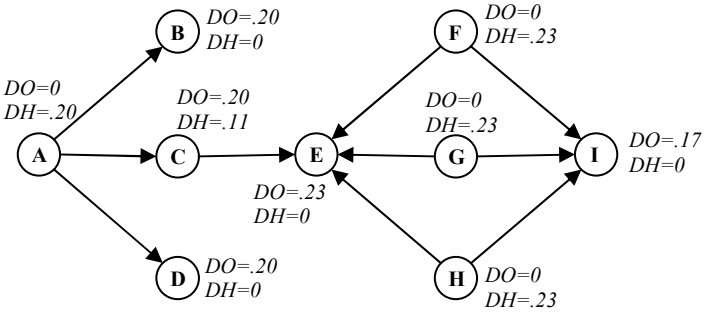


Figure 1.3 - Degrés d'autorité (DO) et d'hubité (DH) calculés par SALSA pour les nœuds du graphe *G3*

Nous remarquons que la matrice $\mathbf{O}_{SALSA}$ n'est pas irréductible. Son graphe contient en effet deux composants connexes à savoir $C_1 = \{\mathrm{B}, \mathrm{C}, \mathrm{D}\}$ et $C_2 = \{\mathrm{E}, \mathrm{I}\}$. Contrairement à HITS qui s'était focalisé sur la composante *C1*, SALSA attribue des degrés d'autorité à tous les nœuds ayant au moins un lien entrant. Il assigne également un degré d'hubité à tous les nœuds qui possèdent au moins un lien sortant. La figure 1.17 montre que le nœud E est la meilleure

autorité et que les nœuds F, G et H sont les meilleurs hubs. Nous constatons aussi que le nœud B, qui n'a qu'un seul lien entrant, possède un degré d'autorité plus fort que le nœud I qui a trois liens entrants. Ce résultat est dû à la méthode utilisée par SALSA pour combiner les degrés d'importance des différentes composantes lorsque la chaine de Markov n'est pas irréductible. Cette méthode bien que simple n'est basée que sur la taille de la composante et ne prend pas en compte la densité des liens au sein des composantes ; elle tend par conséquent à favoriser les nœuds qui appartiennent à des composantes de grande taille.

Hormis le fait que l'algorithme SALSA souffre lui aussi (mais à un degré moindre que HITS) du problème de la TKC, comme le montrent Borodin et al. [Borodin et al. 05] dans leur étude expérimentale, la principale critique que nous pouvons faire à SALSA est le fait qu'il soit équivalent à la centralité du degré (cf. formules 1.11 et 1.12). Or, celle-ci est une mesure locale qui détermine l'importance d'un nœud en utilisant son voisinage immédiat uniquement. L'algorithme SALSA n'utilise donc pas la structure globale du graphe comme c'est le cas de HITS et PageRank.

1.3.4 HubAvg

HubAvg [Borodin et al. 01] est un autre algorithme d'analyse de liens proposé pour résoudre le problème TKC qui caractérise l'algorithme HITS. Comme ce dernier, HubAvg est basé sur le principe de renforcement mutuel entre les hubs et les autorités. Cependant, les auteurs de HubAvg utilisent une définition différente que celle de Kleinberg pour la notion de bon hub. En effet, alors que HITS considère qu'un bon hub est un nœud qui pointe vers de bonnes autorités, dans HubAvg, un bon hub est un nœud qui pointe *uniquement* vers de bonnes autorités. En d'autres termes, cette définition signifie que les hubs qui pointent vers des nœuds ayant un faible degré d'autorité doivent être pénalisés. Plus précisément, le degré d'hubité d'une page p_i est calculé par HubAvg de la façon suivante :

$$H(p_i) = \frac{1}{|out(p_i)|} \sum_{p_j \in out(p_i)} A(p_j) \tag{1.17}$$

où $out(p_i)$ représente l'ensemble des pages pointées par la page p_i et $A(p_j)$ représente le degré d'autorité de la page p_j. Le degré d'hubité d'une page correspond donc à la moyenne des degrés d'autorité des pages qu'elle pointe. Le degré d'autorité d'une page, quant à lui, est calculé de la même manière que dans l'algorithme HITS i.e. il est égal à la somme des degrés d'hubité des pages qui pointent vers elle.

Les différentes étapes de l'algorithme HubAvg sont indiquées par l'algorithme 1.5. Nous remarquons que l'algorithme est similaire à la fois à HITS pour l'étape de mise à jour du vecteur d'autorité (ligne *8*) et à SALSA pour l'étape de mise à jour du vecteur d'hubité (ligne *9*). Notons aussi que le vecteur d'autorité $\mathbf{o}$ (resp. d'hubté $\mathbf{h}$) calculé par HubAvg correspond au vecteur propre principal de la matrice $\mathbf{A}^T\mathbf{A}_l$ (resp. $\mathbf{A}_l\mathbf{A}^T$), où $\mathbf{A}_l$ est la matrice d'adjacence dont les lignes ont été normalisées pour que leur somme soit égale à un.

Afin de voir l'effet de la normalisation introduite dans l'algorithme HubAvg, considérons à nouveau les graphes jouet *G2* et *G3* des figures 1.18 et 1.19 où sont indiqués les résultats obtenus en appliquant l'algorithme HubAvg avec ces deux graphes. Pour le graphe *G2*, nous

remarquons que les résultats de HubAvg sont différents de ceux obtenus avec l'algorithme HITS. En effet, alors que ce dernier (cf. figure 1.15) trouve que les nœuds A et B sont plus importants en termes de degré d'hubité que les nœuds F et G, l'algorithme HubAvg trouve le contraire. Cette différence s'explique par le fait que les nœuds F et G qui possèdent deux liens sortants sont moins pénalisés par HubAvg que les nœuds A et B qui possèdent trois liens sortants. D'autre part, comme les nœuds F et G sont de meilleurs hubs que les nœuds A et B, il en résulte que le nœud H, pointé par F et G, est plus important en termes de degré d'autorité que le nœud E pointé par A et B.

Concernant le graphe *G3*, nous remarquons que HubAvg attribue au nœud C un degré d'hubité plus fort qu'aux nœuds F, G et H alors que HITS (cf. figure 1.16) assigne moins d'importance au nœud C en termes de degré d'hubité. N'ayant qu'un seul lien sortant, le nœud C est en fait moins pénalisé par HubAvg que les nœuds F, G et H qui ont trois liens sortants. De même qu'avec l'algorithme HITS, nous remarquons que l'algorithme HubAvg semble "ignorer" les nœuds A, B et D puisqu'il leur assigne des degrés d'autorité et d'hubité nuls. En effet, tel que défini, l'algorithme HubAvg ne gère pas le cas où le graphe associé à la matrice $\mathbf{A}^T\mathbf{A}_l$ (ou $\mathbf{A}\mathbf{A}_l{}^T$) contient plusieurs composantes connexes ce qui le rend vulnérable au problème de la TKC.

Algorithme 1.5 : L'algorithme HubAvg

Entrée : - une matrice d'adjacence $\mathbf{A} \in \mathbb{R}^{N\times N}$
Sortie : vecteurs d'autorité $\mathbf{o}$ et d'hubité $\mathbf{h}$
début
1. $\mathbf{D}_l \leftarrow diag\left(\mathbf{A}\times\mathbf{1}_{N\times 1}\right)$ // $\mathbf{1}_{N\times 1}$ est un vecteur colonne de dimension N contenant des uns
2. **pour** $i = 1,\ldots,N$ **faire**
3. | **si** $\left(d_l\right)_{ii} = 0$ **alors** $\left(d_l\right)_{ii} \leftarrow 1$ // Permet de rendre la matrice $\mathbf{D}_l$ inversible
4. **fin**
5. $\mathbf{A}_l \leftarrow \mathbf{D}_l{}^{-1}\mathbf{A}$
6. $\mathbf{o}^{(0)} \leftarrow \mathbf{1}_{N\times 1}$, $\mathbf{h}^{(0)} \leftarrow \mathbf{1}_{N\times 1}$, $t \leftarrow 1$
7. **répéter**
8. | $\mathbf{o}^{(t)} \leftarrow \mathbf{A}^T\mathbf{h}^{(t-1)}$
9. | $\mathbf{h}^{(t)} \leftarrow \mathbf{A}_l\mathbf{o}^{(t-1)}$
10. | $\mathbf{o}^{(t)} \leftarrow \frac{\mathbf{o}^{(t)}}{\left\|\mathbf{o}^{(t)}\right\|_2}$, $\mathbf{h}^{(t)} \leftarrow \frac{\mathbf{h}^{(t)}}{\left\|\mathbf{h}^{(t)}\right\|_2}$
11. | $t \leftarrow t+1$
12. **jusqu'à** *convergence*

fin

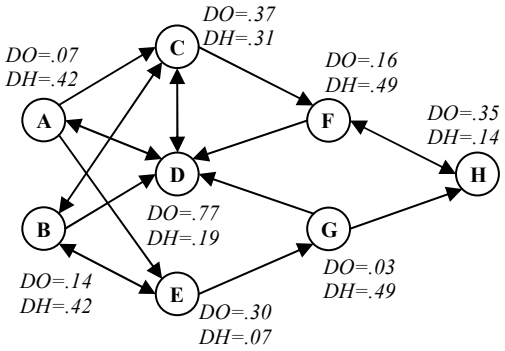


Figure 1.4 – Degrés d'autorité (DO) et d'hubité (DH) calculés par HubAvg pour les nœuds du graphe G2

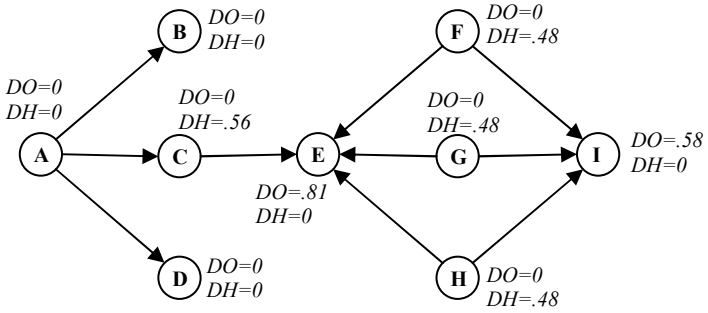


Figure 1.5 - Degrés d'autorité (DO) et d'hubité (DH) calculés par HubAvg pour les nœuds du graphe G3

L'algorithme HubAvg possède les mêmes inconvénients que l'algorithme HITS. Ainsi, si la matrice $\mathbf{A}^T\mathbf{A}_1$ (ou $\mathbf{A}\mathbf{A}_1^T$) n'est pas irréductible, l'existence et l'unicité de son vecteur propre principal pose problème comme pour l'algorithme HITS. De plus, l'algorithme HubAvg souffre lui aussi du problème de la TKC mais de manière moins importante grâce à la normalisation utilisée lors du calcul des degrés d'hubité [Borodin et al. 05].

1.4 Bilan

Dans ce chapitre, nous avons présenté diverses définitions de la notion de centralité dans les graphes. Nous avons commencé par la plus simple de ces définitions, à savoir la centralité de degré, qui correspond à la taille du voisinage immédiat d'un nœud. La centralité de degré est bien connue en bibliométrie où le degré entrant (appelé aussi nombre de citations) est utilisé pour quantifier l'importance des publications scientifiques [Garfield 72]. Cependant, un fort degré entrant n'est généralement pas suffisant pour repérer tous les documents importants [Maslov and Redner 08]. En effet, nous montrerons dans le chapitre suivant que si le graphe de documents contient plusieurs communautés (où chacune correspond à une thématique par exemple), alors la centralité de degré favorise les documents appartenant aux

communautés qui sont de grande taille et fortement connectées. La centralité de degré est en fait une mesure locale qui ne tient pas compte de la structure globale du graphe.

Trois autres mesures de centralité prenant en compte la structure globale du graphe ont ensuite été présentées. Il s'agit des centralités de proximité, d'intermédiarité et de vecteur propre. Nous avons montré que ces trois mesures ne sont pas adaptées à l'analyse de graphes de documents car ces derniers sont orientés et généralement non connexes.

Nous avons également décrit en détail les principaux algorithmes de calcul de centralité dans les graphes de documents. L'analyse des avantages et inconvénients de ces algorithmes a mis en évidence un problème important dont souffrent ces techniques. Il s'agit en l'occurrence du problème de la TKC (Tightly Knit Community) qui entraine une attribution erronée des degrés d'importance aux documents.

Par ailleurs, nous avons établi un lien entre les mesures issues de l'ARS et celles issues de la RI. Il provient du fait que les mesures issues de la RI sont en réalité toutes basées sur le principe de la centralité spectrale (ou de vecteur propre) introduite au début des année 1970 par Bonacich [Bonacich 72].

Notre objectif, dans le chapitre suivant, est de développer une mesure de centralité dans les graphes de documents qui n'utilise que les liens entre documents, qui exploite la structure globale du graphe et qui soit robuste au problème de la TKC. Nous allons ainsi proposer trois nouveaux algorithmes de calcul de centralité qui permettent de calculer les degrés d'autorité et d'hubité des documents tout en évitant l'effet TKC dont souffrent les précédents algorithmes.

2

Nouveaux Algorithmes pour le Calcul de Centralité dans les Graphes de Documents

Dans le chapitre précédent, nous avons décrit brièvement l'effet TKC qui caractérise l'algorithme HITS. Dans le présent chapitre, nous allons nous intéresser plus en détail à ce problème en identifiant notamment ses différentes causes. Nous pensons en effet que le "symptôme" TKC est un paramètre important qui doit être pris en compte par les algorithmes de calcul de centralité.

Nous proposons dans ce chapitre trois nouveaux algorithmes de calcul de centralité dans les graphes de documents à savoir MHITS, NHITS et DocRank. Ces algorithmes calculent pour chaque document un degré d'autorité ainsi qu'un degré d'hubité. De plus, chaque algorithme utilise une stratégie différente pour pallier au problème de la TKC.

Afin d'évaluer les trois algorithmes proposés, nous avons mené des expérimentations dans lesquelles nous avons comparé onze algorithmes de calcul de centralité (y compris les nôtres) en utilisant huit graphes de documents. Les résultats obtenus montrent l'intérêt des algorithmes proposés et notamment de l'algorithme DocRank qui donne des résultats largement meilleurs à ceux des autres algorithmes. De plus, un avantage non négligeable de DocRank est qu'il possède une complexité très faible. En effet, contrairement à la plupart des autres algorithmes qui sont basés sur le calcul de vecteurs propres (qui peut être coûteux en temps de calcul lorsque le graphe est de grande taille), DocRank a la même complexité que la centralité de degré.

2.1 L'effet TKC (Tightly Knit Community)

L'effet TKC est sans doute l'une des principales raisons qui ont fait que l'algorithme HITS n'a pas connu un aussi grand succès que l'algorithme PageRank. Comme ce dernier, l'algorithme HITS a été proposé initialement en recherche d'information pour le classement ("ranking") de documents sur le web. Dans sa version "complète", HITS [Kleinberg 99a] construit dans un premier temps un graphe de documents à partir d'une requête utilisateur (cette étape est décrite dans la section 2.5.1), puis dans un deuxième temps, il effectue le calcul des degrés d'autorité et d'hubité. Plusieurs chercheurs (par exemple [Borodin et al. 05][Lempel and Moran 00][Cohn and Chang 00]) ayant étudié l'algorithme HITS ont remarqué que les documents classés comme importants ne traitent en général qu'une seule thématique de la requête utilisateur. Pire encore, il a même été observé que dans certains cas, HITS classe comme importants des documents qui ne sont pas du tout pertinents par rapport à la requête initiale (problème connu sous le nom de *topic drift* [Bharat and Henzinger 98]).

La figure 2.1 montre le classement des 10 documents les plus importants obtenu en appliquant l'algorithme HITS avec un graphe d'environ 5000 publications scientifiques dans le domaine de la physique des plasmas. Pour ce même graphe, la figure 2.2 indique les 10 documents les plus populaires retournés par l'algorithme PageRank. Sans avoir à consulter les contenus des documents, nous pouvons remarquer que les articles classés comme étant les plus importants (en termes d'autorité) par HITS relèvent tous de la thématique « dusty plasma » ; presque tous les documents retournés contiennent en effet ce terme dans leur titre. Les 10 articles les plus importants selon PageRank semblent être, quant à eux, beaucoup plus diversifiés en termes de thématiques traitées.

1- *Dust*-acoustic waves in *dusty plasmas*
2- *Dusty plasmas* in the solar system
3- Laboratory observation of the *dust*-acoustic wave mode
4- Plasma crystal: Coulomb crystallization in a *dusty plasma*
5- Direct observation of Coulomb crystals and liquids in strongly coupled rf *dusty plasmas*
6- *Dust* ion-acoustic wave
7- Cosmic *Dusty Plasmas*
8- The electrostatics of a *dusty plasma*
9- Laboratory studies of waves and instabilities in *dusty plasmas*
10- Condensed Plasmas under Microgravity

Figure 2.1 - Les 10 documents les plus importants d'après l'algorithme HITS

1- Centrifugally driven diffusion of Iogenic plasma
2- Factors governing the ratio of inward to outward diffusing flux of satellite ions
3- Helical microtubules of graphitic carbon
4- N-dependence in the classical one-component plasma Monte Carlo calculations
5- A General Formula for the Estimation of Dielectronic Recombination Co-Efficients in …
6- Ionization Equilibrium and Radiative Cooling of a Low-Density Plasma
7- Radiative cooling of a low-density plasma
8- A survey of the plasma electron environment of Jupiter - A view from Voyager
9- Strong turbulence of plasma waves
10- A General Theory of the Plasma of an Arc

Figure 2.2 - Les 10 documents les plus importants d'après l'algorithme PageRank

Soit *G* un graphe de documents représenté par sa matrice d'adjacence **A**. Nous noterons par G_o (resp. G_h) le graphe associé à la matrice $\mathbf{O}_{HITS} = \mathbf{A}^T\mathbf{A}$ (resp. $\mathbf{H}_{HITS} = \mathbf{A}\mathbf{A}^T$). G_o et G_h sont par définition des graphes non-orientés car leurs matrices d'adjacence sont symétriques. Les propriétés des graphes G_o et G_h ont été étudiées dans plusieurs travaux (par exemple [Ding et al. 02]). Parmi ces propriétés, nous citerons le fait qu'ils possèdent le même nombre de composantes connexes et le fait que leurs matrices d'adjacence respectives possèdent les mêmes valeurs propres. Rappelons également que le vecteur d'autorité (resp. d'hubité) calculé par HITS correspond au vecteur propre principal de la matrice d'adjacence du graphe G_o (resp. G_h).

Considérons à présent le graphe jouet *G1* de la figure 2.3a. Le graphe d'autorité $G1_o$ est indiqué par la figure 2.3b (bien que nous nous intéressions dans ce qui suit au graphe d'autorité uniquement, ce que nous présentons s'applique par analogie au graphe d'hubité). Nous remarquons que le graphe $G1_o$ n'est pas connexe puisqu'il contient deux composantes connexes à savoir $C1o = \{\mathrm{D},\mathrm{E},\mathrm{F}\}$ et $C2o = \{\mathrm{J},\mathrm{K},\mathrm{L}\}$. Les degrés d'autorité et d'hubité des nœuds du graphe *G1* sont indiqués sur le tableau 2.1. Celui-ci indique que les nœuds D, E et F ont un degré d'autorité égal à 0.33 et que tous les autres nœuds ont un degré d'autorité nul. Les nœuds J, K et L possèdent pourtant tous les trois des liens entrants. On peut ainsi dire que l'algorithme HITS a attribué la totalité (i.e. 100%) de l'autorité aux nœuds de la composante *C1o*. Il s'agit là du premier symptôme de l'effet TKC qui est causé par la non connexité du graphe d'autorité. En effet, si le graphe d'autorité contient plusieurs composantes connexes, l'algorithme HITS se focalise sur une seule de ces composantes. Les nœuds appartenant aux autres composantes obtiennent alors tous un degré d'autorité nul. Lors de nos expérimentations avec différents graphes, nous avons remarqué que la composante connexe sur laquelle se focalise HITS ne correspond pas forcément à celle qui possède le plus de nœuds ou à celle qui contient le plus de liens. Nous avons remarqué que la composante qui absorbe la totalité de l'autorité correspond à celle dont la valeur propre principale est la plus grande.

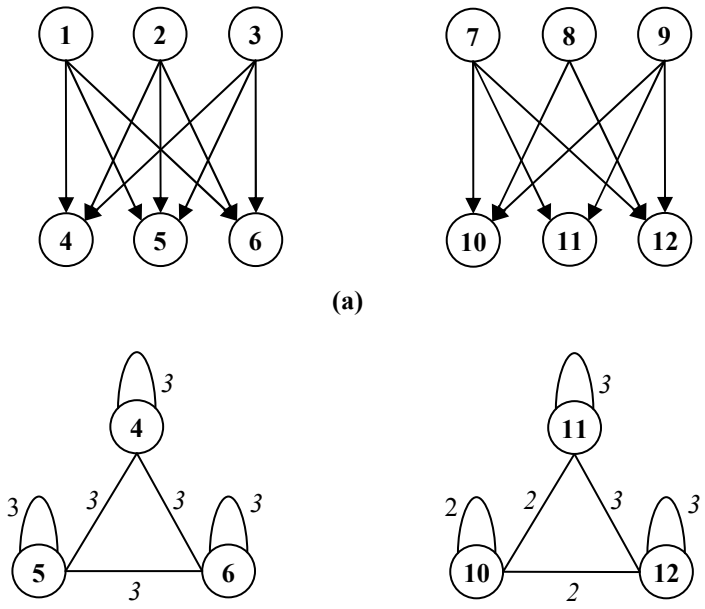


Figure 2.3 - (a) Graphe jouet *G1*
(b) Graphe $G1_o$ (graphe d'autorité construit à partir du graphe *G1*)

Tableau 2.1 - Degrés d'autorité et d'hubité calculés par HITS pour les nœuds du graphe *G1*

Nœud	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*
Degré d'autorité	0	0	0	0.33	0.33	0.33	0	0	0	0	0	0
Degré d'hubité	0.33	0.33	0.33	0	0	0	0	0	0	0	0	0

Cependant, l'effet TKC ne se produit pas uniquement lorsque le graphe d'autorité (ou d'hubité) n'est pas connexe. Pour illustrer cela, considérons les graphes jouets *G2* de la figure 2.4a et *G3* de la figure 2.5a dont les graphes d'autorité respectifs $G2_o$ et $G3_o$ sont connexes. Les graphes *G2* et *G3* diffèrent par l'absence dans ce dernier d'un lien entre les nœuds *14* et *17*. Comme l'indique le tableau 2.2 (resp. le tableau 2.3) cette différence bien que légère a néanmoins des conséquences très significatives sur les degrés d'autorité (resp. d'hubité) attribués aux nœuds de ces deux graphes. En effet, nous remarquons qu'avec le graphe *G2*, les nœuds 16, 17 et 18 obtiennent à eux trois 40% de l'autorité totale tandis que ces mêmes nœuds n'obtiennent que 1% de l'autorité totale dans le graphe *G3*. Les nœuds 4, 5 et 6 totalisent quant à eux plus de 90% de l'autorité dans le graphe *G3* ! Ce "mauvais comportement" est dû au fait que l'algorithme HITS se focalise sur un sous-graphe particulier du graphe d'autorité (ou d'hubité) qui forme une structure appelée *communauté*. Nous considérerons dans ce chapitre une communauté comme étant un sous-graphe connexe contenant beaucoup de liens internes (i.e. des liens entre les nœuds du sous-graphe) et peu de liens externes (i.e. des liens avec des

nœuds qui n'appartiennent pas au sous-graphe). La notion de communauté fera l'objet de la deuxième partie de cette thèse.

Ainsi, l'algorithme HITS semble considérer que le graphe $G2_o$ contient trois communautés à savoir $S1o = \{4,5,6\}$, $S2o = \{10,11\}$ et $S3o = \{16,17,18\}$ et alloue la majorité de l'autorité (92%) aux nœuds des communautés $S1o$ et $S2o$. Ces deux communautés obtiennent le même degré d'autorité total (46%) car elles sont identiques. Avec le graphe *G3*, la situation est très différente. L'algorithme HITS attribue un fort degré d'autorité aux nœuds de la communauté $S1o$ et un très faible degré d'autorité aux nœuds de la communauté $S3o$. HITS considère, en fait, que la communauté $S1o$ est la "plus importante" et attribue par conséquent la majorité de l'autorité aux nœuds de cette communauté.

La deuxième cause de l'effet TKC est donc la présence dans le graphe d'autorité (ou d'hubité) de groupes de nœuds fortement connectés entre eux, qui forment ce que l'on appelle des communautés.

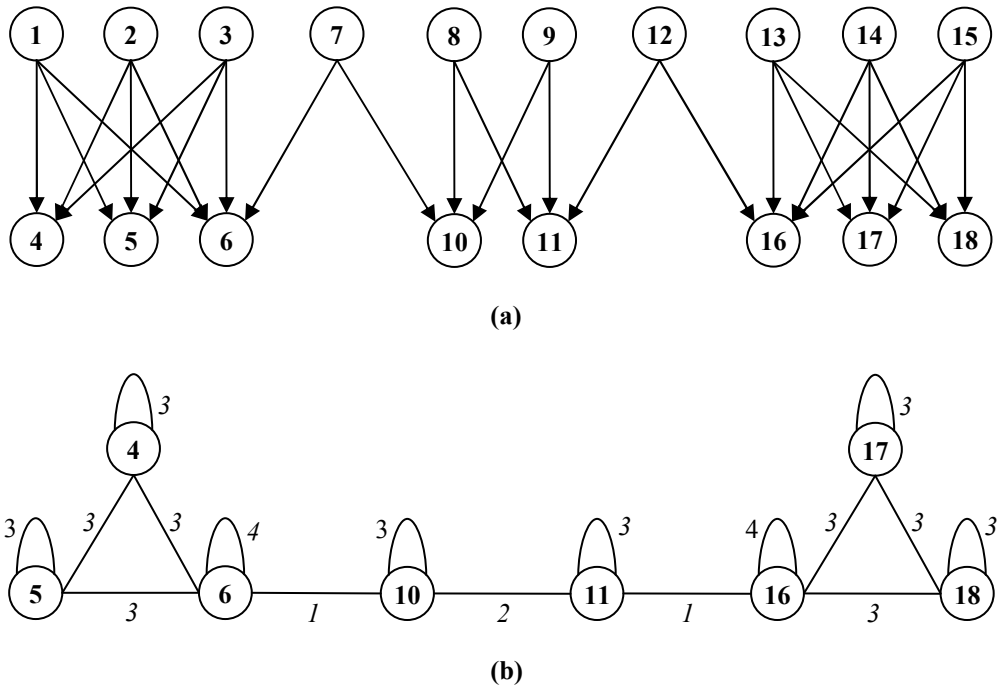
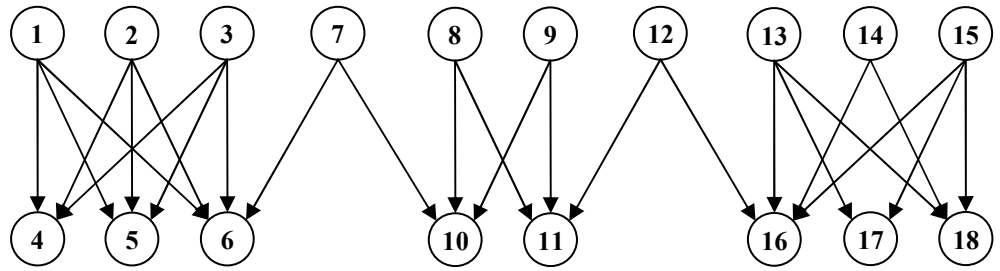
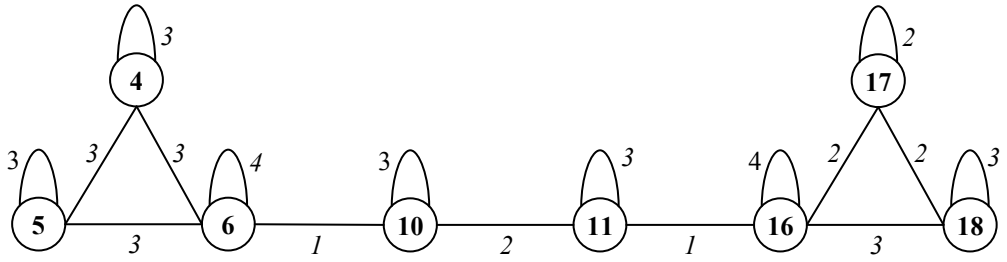


Figure 2.4 – (a) Graphe jouet *G2* ;
(b) Graphe $G2_o$ (graphe d'autorité construit à partir du graphe G2)



(a)



(b)

Figure 2.5 - Graphe jouet G3 ;
(b) Graphe $G3_o$ (graphe d'autorité construit à partir du graphe G3)

Tableau 2.2 - Degrés d'autorité calculés par HITS pour les nœuds des graphes *G2* et *G3*

Nœud(s)	*1,2,3,7,8,9,12,13,14,15*	*4*	*5*	*6*	*10*	*11*	*16*	*17*	*18*
Graphe *G2*	0	0.14	0.14	0.17	0.04	0.04	0.17	0.14	0.14
Graphe *G3*	0	0.29	0.29	0.33	0.06	0.02	0.01	0	0

Tableau 2.3 - Degrés d'hubité calculés par HITS pour les nœuds des graphes *G2* et *G3*

Nœud(s)	*4,5,6,10,11,16,17,18*	*1*	*2*	*3*	*7*	*8*	*9*	*12*	*13*	*14*	*15*
Graphe *G2*	0	0.14	0.14	0.14	0.06	0.02	0.02	0.06	0.14	0.14	0.14
Graphe *G3*	0	0.27	0.27	0.27	0.12	0.02	0.02	0.01	0	0	0

2.2 L'algorithme MHITS (Multi-HITS)

2.2.1 Principe

Le premier algorithme que nous proposons est une variante de l'algorithme HITS qui traite la première cause de l'effet TKC. Le but de l'algorithme MHITS est de s'assurer que tous les documents obtiennent un degré d'autorité et/ou d'hubité strictement positif quelque soit la composante connexe à laquelle ils appartiennent. En effet, dans le cas où le graphe d'autorité G_o (resp. d'hubité G_h) contient plusieurs composantes connexes, HITS n'assigne des degrés d'autorité (resp. d'hubité) qu'aux nœuds appartenant à une seule de ces composantes connexes. Concrètement, l'algorithme MHITS calcule les degrés d'autorité et d'hubité des documents en appliquant les trois étapes suivantes :

i) Identifier les composantes connexes C^o (resp. C^h) du graphe G_o (resp. G_h). Les graphes G_o et G_h étant non-orientés, nous proposons alors d'utiliser l'algorithme de parcours en largeur (*Breadth-first search* ou *BFS*) pour calculer en un temps linéaire les composantes connexes de ces graphes [Bornholdt and Schuster 03].

ii) Calculer le vecteur propre principal $\mathbf{x}_i$ (resp. $\mathbf{y}_i$) pour chaque composante C_i^o (resp. C_i^h). Chaque vecteur $\mathbf{x}_i$ (resp. $\mathbf{y}_i$) est ensuite normalisé de telle sorte que $\sum_j x_{ij} = 1$ (resp. $\sum_j y_{ij} = 1$). Cette normalisation signifie que toutes les composantes possèdent une autorité (resp. hubité) totale égale à 1.

iii) Calculer le degré d'autorité (resp. d'hubité) final de chaque nœud. Celui-ci est obtenu en multipliant le degré d'autorité (resp. d'hubité) de chaque nœud dans la composante à laquelle il appartient par un facteur qui reflète l'importance de cette composante. Nous avons choisi ici de quantifier l'importance d'une composante par la proportion de nœuds du graphe appartenant à cette composante. Cette méthode de pondérer les degrés d'importance des nœuds est similaire à celle utilisée par Lempel et Morin [Lempel and Moran 00] dans l'algorithme SALSA. Concernant cette pondération des composantes, on pourrait également utiliser le nombre de liens au sein de la composante ou encore la valeur propre principale associée au vecteur d'autorité (ou d'hubité) de la composante.

Lorsque le graphe G_o (ou G_h) est connexe, il est évident que l'algorithme MHITS est équivalent à l'algorithme HITS. Par ailleurs, l'inconvénient majeur de l'algorithme MHITS est qu'il ne résout pas la deuxième cause de l'effet TKC. Ainsi, si les composantes connexes du graphe G_o (resp. G_h) contiennent plusieurs communautés, les nœuds n'appartenant pas à la communauté "la plus importante" seront "défavorisés" car l'algorithme leur attribuera un degré d'autorité (resp. d'hubité) très faible.

2.2.2 Détails de l'algorithme

Les différentes étapes de l'algorithme MHITS sont décrites par l'algorithme 2.1.

Algorithme 2.1 : L'algorithme MHITS (Multi-HITS)

Entrée : - un graphe $G=(V,E)$ d'ordre N représenté par sa matrice d'adjacence $\mathbf{A}$

Sortie : vecteurs d'autorité $\mathbf{o}$ et d'hubité $\mathbf{h}$

début

1. // En utilisant la méthode du parcours en largeur (BFS), calculer les K composantes
// connexes des graphes non-orientés G_o et G_h représentés respectivement par les
// matrices $\mathbf{A}^T\mathbf{A}$ et $\mathbf{A}\mathbf{A}^T$:

$$\left\{C_1^o,\ldots,C_K^o\right\} \leftarrow BFS\left(\mathbf{A}^T\mathbf{A}\right)$$

$$\left\{C_1^h,\ldots,C_K^h\right\} \leftarrow BFS\left(\mathbf{A}\mathbf{A}^T\right)$$

2. // En utilisant la méthode des puissances (PM), calculer pour chaque composante
// C_k^o (resp. C_k^h) son vecteur propre dominant $\mathbf{x}_k$ (resp. $\mathbf{y}_k$) :

pour $k=1\ldots K$ **faire**

$$\mathbf{x}_k \leftarrow PM\left(C_k^o\right)$$

$$\mathbf{y}_k \leftarrow PM\left(C_k^h\right)$$

$$\mathbf{x}_k \leftarrow \frac{\mathbf{x}_k}{\left\|\mathbf{x}_k\right\|_1} \;;\; \mathbf{y}_k \leftarrow \frac{\mathbf{y}_k}{\left\|\mathbf{y}_k\right\|_1} \;\text{// } \left\|\mathbf{v}\right\|_1=\sum_{i=1} v_i \text{ est la norme L1 du vecteur } \mathbf{v}$$

fin

3. // Calculer le vecteur d'autorité $\mathbf{o}$ et d'hubité $\mathbf{h}$:

$\mathbf{o} \leftarrow \mathbf{0}_{N\times 1}$, $\mathbf{h} \leftarrow \mathbf{0}_{N\times 1}$ // $\mathbf{0}_{N\times 1}$ est un vecteur colonne de dimension N contenant des zéros

pour $k=1\ldots K$ **faire**

  pour $l=1,\ldots,\left|C_k^o\right|$ **faire**

$$j \leftarrow C_k^o(l)$$

$$o_j \leftarrow \left|C_k^o\right| x_{kl}$$

  fin

  pour $l=1,\ldots,\left|C_k^h\right|$ **faire**

$$j \leftarrow C_k^h(l)$$

$$h_j \leftarrow \left|C_k^h\right| y_{kl}$$

  fin

fin

4. // Normalisation des vecteurs $\mathbf{o}$ et $\mathbf{h}$:

$$\mathbf{o} \leftarrow \frac{\mathbf{o}}{\left\|\mathbf{o}\right\|_1} \;;\; \mathbf{h} \leftarrow \frac{\mathbf{h}}{\left\|\mathbf{h}\right\|_1}$$

fin

2.2.3 Exemples jouets

Les vecteurs d'autorité et d'hubité obtenus en appliquant l'algorithme MHITS avec le graphe jouet *G1* sont indiqués par le tableau 2.4. Nous remarquons que contrairement à l'algorithme HITS qui n'attribue des degrés d'autorité (resp. d'hubité) qu'aux nœuds 4, 5 et 6 (resp. 1, 2 et 3), l'algorithme MHITS attribue des degrés d'autorité (d'hubité) à tous les nœuds ayant des liens entrants (resp. sortants). L'algorithme MHITS permet donc d'éviter l'effet TKC lorsque celui-ci est dû à non connexité du graphe d'autorité (ou d'hubité).

Avec le graphe jouet *G3*, MHITS donne les mêmes résultats que l'algorithme car le graphe d'autorité $G3_o$ ne contient qu'une seule composante. MHITS souffre lui aussi de l'effet TKC avec ce graphe.

Tableau 2.4 - Degrés d'autorité et d'hubité calculés par MHITS pour les nœuds du graphe *G1*

Nœud	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*
Degré d'autorité	0	0	0	0.17	0.17	0.17	0	0	0	0.18	0.13	0.18
Degré d'hubité	0.17	0.17	0.17	0	0	0	0.18	0.13	0.18	0	0	0

2.2.4 Convergence de l'algorithme

Hormis le fait de traiter la première cause de l'effet TKC, MHITS possède un autre avantage par rapport à HITS concernant la convergence. La convergence de l'algorithme HITS pose en effet problème lorsque la plus grande valeur propre de la matrice d'autorité (i.e. matrice d'adjacence du graphe d'autorité) se répète [Farahat et al. 05]. Une telle situation est illustrée par le graphe jouet *G4* de la figure 2.6 et dont les vecteurs d'autorité et d'hubité calculés par HITS sont indiqués par le tableau 2.5. La matrice d'autorité du graphe *G4* possède deux valeurs propres : $\lambda_1 = 3$ et $\lambda_2 = 3$. La valeur propre 3 est donc double, ce qui explique les résultats "étranges" (nœuds ayant un degré d'autorité ou d'hubité négatif) du tableau 2.5. Ces résultats sont en contradiction avec le théorème de Perron-Frobenius selon lequel le vecteur propre principal d'une matrice symétrique définie positive (comme c'est le cas des matrices d'autorité et d'hubité) est toujours positif.

L'algorithme MHITS quant à lui n'est pas concerné par ce problème de convergence comme le montrent par exemple les résultats du tableau 2.6. MHITS traite en effet chaque composante connexe du graphe d'autorité (et d'hubité) de manière indépendante.

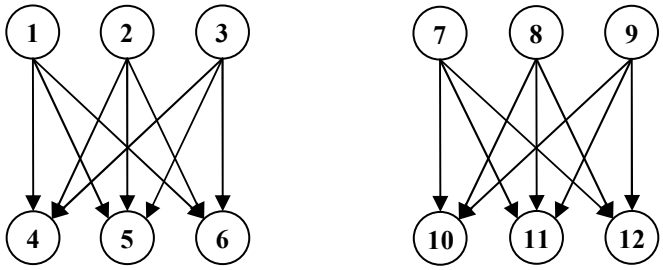


Figure 2.6 - Graphe jouet *G4*

Tableau 2.5 - Degrés d'autorité et d'hubité calculés par HITS pour les nœuds du graphe *G4*

Nœud	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*
Degré d'autorité	0	0	0	0.43	0.43	0.43	0	0	0	-0.1	-0.1	-0.1
Degré d'hubité	0.43	0.43	0.43	0	0	0	-0.1	-0.1	-0.1	0	0	0

Tableau 2.6 - Degrés d'autorité et d'hubité calculés par MHITS pour les nœuds du graphe *G4*

Nœud	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*
Degré d'autorité	0	0	0	0.16	0.16	0.16	0	0	0	0.16	0.16	0.16
Degré d'hubité	0.16	0.16	0.16	0	0	0	0.16	0.16	0.16	0	0	0

2.3 L'algorithme NHITS (Non-Negative HITS)

2.3.1 HITS et la décomposition en valeurs singulières

Dans l'article [Kleinberg 99a], Kleinberg évoque le fait que les vecteurs propres secondaires (par opposition au vecteur propre principal) de la matrice $\mathbf{A}^T\mathbf{A}$ (resp. $\mathbf{A}\mathbf{A}^T$) peuvent être utilisés pour obtenir les degrés d'autorités et d'hubité relatifs à d'autres communautés (i.e. des communautés secondaires autres que la principale). Cette idée proposée par Kleinberg n'est cependant pas satisfaisante en pratique car les vecteurs d'autorité et d'huibté secondaires peuvent contenir des valeurs positives et des valeurs négatives ce qui rend leur interprétation délicate. Le théorème de Perron-Frobenius nous dit en effet que seul le vecteur propre principal d'une matrice symétrique définie positive est positif. Si l'on considère le graphe *G5* de la figure 2.7 et que l'on calcule les deux vecteurs propres de la matrice d'autorité $\mathbf{A}^T\mathbf{A}$ associés aux deux plus grandes valeurs propres, nous obtenons les résultats indiqués par le tableau 2.7.

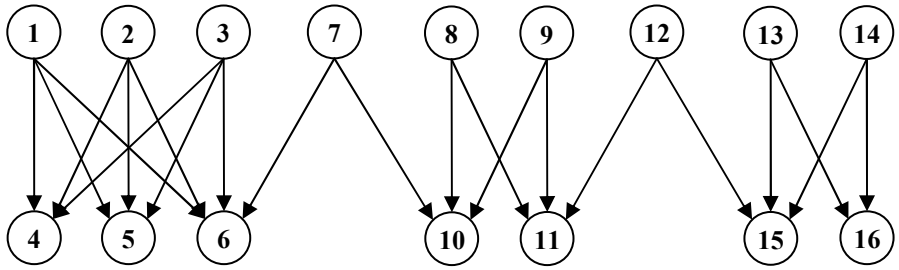


Figure 2.7 - Graphe jouet G5

Tableau 2.7 – Premier et second vecteurs propres de la matrice $G5_o$ (matrice d'autorité associée à *G5*)

Nœud	*1,2,3,7,8,9,12,13,14*	*4*	*5*	*6*	*10*	*11*	*15*	*16*
Premier vecteur propre	0	0.55	0.55	0.62	0.11	0.03	0.01	0
Second vecteur propre	0	0.08	0.08	-0.02	-0.52	-0.63	-0.49	-0.29

Nous remarquons que le premier vecteur ne contient que des valeurs positives alors que le deuxième vecteur contient à la fois des valeurs positives et des valeurs négatives. La présence de valeurs ayant des signes différents rend l'interprétation des vecteurs propres secondaires difficile car cela nécessiterait de les inspecter manuellement, sachant que dans certains cas c'est la partie négative qui est pertinente alors que dans d'autres cas c'est la partie positive qui l'est.

Le calcul de plusieurs (*K*) vecteurs propres des matrices $\mathbf{A}^T\mathbf{A}$ et $\mathbf{A}\mathbf{A}^T$ revient en fait à effectuer une décomposition en valeurs singulières (ou SVD : Singular Value Decomposition) de la matrice d'adjacence **A** [Chikhi et al. 07]. Cette décomposition est donnée par [Golub and Van Loan 96]:

$$\mathbf{A} = \mathbf{U}_{\pm}\mathbf{S}_{+}\mathbf{V}_{\pm} + \mathbf{B}_1 \tag{2.1}$$

où : - $\mathbf{U}_{\pm}$ est une matrice de dimensions $N \times K$ contenant les *K* vecteurs propres de la matrice d'hubité $\mathbf{A}\mathbf{A}^T$. Cette matrice contient des valeurs positives et négatives.

- $\mathbf{S}_{+}$ est une matrice diagonale de dimensions $K \times K$ contenant les valeurs propres par ordre décroissant. Ces valeurs sont toujours positives car les matrices $\mathbf{A}^T\mathbf{A}$ et $\mathbf{A}\mathbf{A}^T$ sont symétriques définies positives.

- $\mathbf{V}_{\pm}$ est une matrice de dimensions $K \times N$ contenant les *K* vecteurs propres de la matrice d'autorité $\mathbf{A}^T\mathbf{A}$. Cette matrice contient des valeurs positives et négatives.

- $\mathbf{B}_1$ est une matrice de dimension $N \times N$ qui représente un modèle de bruit gaussien ("Gaussian noise model").

Cette décomposition consiste en fait à trouver trois matrices **U**, **S** et **V** tel que la distance euclidienne entre la matrice **A** et la matrice **USV** soit minimale [Golub and Van Loan 96].

L'équation (2.1) peut également être réécrite de la manière suivante :

$$\mathbf{A} = \mathbf{X}_{\pm}\mathbf{Y}_{\pm} + \mathbf{B}_1 \tag{2.2}$$

où : - $\mathbf{X}_{\pm} = \mathbf{U}_{\pm}\left(\mathbf{S}_{+}\right)^{1/2}$ est une matrice indiquant le degré d'hubité des documents dans chacune des *K* communautés.

- $\mathbf{Y}_{\pm} = \left(\mathbf{S}_{+}\right)^{1/2}\mathbf{V}_{\pm}$ est une matrice indiquant le degré d'autorité des documents dans chacune des *K* communautés.

La SVD d'une matrice **A** peut donc être vue comme un problème d'optimisation où l'on cherche deux matrices **X** et **Y** contenant des valeurs positives et des valeurs négatives telles que la distance euclidienne entre **A** et **XY** soit minimale.

2.3.2 Décomposition de la matrice d'adjacence en matrices non-négatives

L'algorithme NHITS que nous proposons ici est basé sur l'idée de calculer plusieurs vecteurs d'autorité (resp. d'hubité) puis de les combiner pour obtenir un seul vecteur d'autorité (resp. d'hubité). Pour le calcul de plusieurs vecteurs d'autorité, nous envisageons cette tâche comme un problème d'optimisation qui est proche de la SVD mais avec une contrainte supplémentaire à savoir la positivité des matrices de la décomposition. Plus précisément, nous cherchons à décomposer la matrice d'adjacence **A** en un produit de deux matrices **X** et **Y** telles que :

$$\mathbf{A} = \mathbf{X}_{+}\mathbf{Y}_{+} + \mathbf{B}_2 \tag{2.3}$$

où : - $\mathbf{X}_{+} \in \mathbb{R}_{+}^{N\times K}$ et $\mathbf{Y}_{+} \in \mathbb{R}_{+}^{K\times N}$ sont des matrices positives et $\mathbf{B}_2 \in \mathbb{R}^{N\times N}$ est un modèle de bruit gaussien.

Les matrices **X** et **Y** sont obtenues en optimisant la fonction objectif suivante :

$$J = \min_{X\geq 0, Y\geq 0} \left\|\mathbf{A} - \mathbf{XY}\right\|_2 \tag{2.4}$$

L'optimisation de (2.4) est en fait similaire au problème de la décomposition en matrices non-négatives (Nonnegative Matrix Factorization) traité par Lee et Seung dans [Lee and Seung 99] et [Lee and Seung 01]. Ces derniers ont proposé un algorithme simple, appelé NMF, permettant de résoudre le problème (2.4). Cet algorithme se résume en deux règles de mise à jour qui sont appliquées de manière itérative jusqu'à ce qu'un critère de convergence soit satisfait. Pour plus d'information concernant la décomposition de la matrice d'adjacence en matrices non-négatives, le lecteur est invité à consulter l'article [Chikhi et al. 08a] dans lequel nous détaillons ce point.

Une fois les matrices **X** (matrice dont les colonnes correspondent aux vecteurs d'hubité) et **Y** (matrice dont les lignes correspondent aux vecteurs d'autorité) calculées, nous combinons les *K* vecteurs d'autorité (resp. d'hubité) de la manière suivante : le vecteur

d'autorité (resp. d'hubité) global est égal à la somme pondérée des *K* vecteurs d'autorité (resp. d'hubité) ; chaque vecteur d'autorité (resp. d'hubité) étant pondéré par un coefficient qui exprime l'importance de la dimension qu'il représente.

2.3.3 Détails de l'algorithme

L'algorithme 2.2 décrit les différentes étapes de l'algorithme NHITS. Concernant le critère de convergence de la décomposition en matrices non-négatives (étape 2), nous considérons qu'il y a convergence lorsque la diminution de la distance euclidienne entre deux itérations est inférieure à un seuil (par exemple 10^{-6}).

2.3.4 Résultats avec les graphes jouets

Le tableau 2.8 indique les degrés d'autorité et d'hubité des nœuds du graphe jouet *G1* en utilisant NHITS avec *K*=2. Nous constatons que NHITS assigne des degrés d'autorité (resp. d'hubité) à tous les nœuds ayant des liens entrants (resp. sortants) malgré la non-connexité du graphe d'autorité $G1_o$.

Le vecteur d'autorité (resp. d'hubité) du graphe jouet *G3* obtenu en utilisant l'algorithme NHITS avec *K*=3 est indiqué par le tableau 2.9 (resp. par le tableau 2.10). Comme avec l'exemple précédent, nous remarquons que l'algorithme attribue des degrés d'autorité (resp. d'hubité) non nuls à tous les nœuds ayant des liens entrants (resp. sortants).

Tableau 2.8 - Degrés d'autorité et d'hubité calculés par NHITS pour les nœuds du graphe *G1*

Nœud	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*
Degré d'autorité	0	0	0	0.17	0.17	0.17	0	0	0	0.17	0.13	0.17
Degré d'hubité	0.17	0.17	0.17	0	0	0	0.17	0.13	0.17	0	0	0

Tableau 2.9 - Degrés d'autorité calculés par NHITS pour les nœuds des graphes *G3*

Nœud(s)	*1,2,3,7,8,9,12,13,14,15*	*4*	*5*	*6*	*10*	*11*	*16*	*17*	*18*
Degré d'autorité	0	0.12	0.12	0.15	0.13	0.13	0.15	0.08	0.12

Tableau 2.10 - Degrés d'hubité calculés par NHITS pour les nœuds des graphes *G3*

Nœud(s)	*4,5,6,10,11,16,17,18*	*1,2,3*	*7*	*8,9*	*12*	*13,15*	*14*
Degré d'hubité	0	0.11	0.09	0.09	0.09	0.11	0.08

Algorithme 2.2 : L'algorithme NHITS (Non-negative HITS)

Entrée : - un graphe $G=(V,E)$ d'ordre N représenté par sa matrice d'adjacence **A**

- K : nombre de dimensions de l'espace de projection

Sortie : vecteurs d'autorité **o** et d'hubité **h**

début

1. // Initialisation des matrices $\mathbf{X} \in \mathbb{R}^{N \times K}$ et $\mathbf{Y} \in \mathbb{R}^{K \times N}$ par des valeurs aléatoires positives :

   $\mathbf{X} \leftarrow rand(N,K)$

   $\mathbf{Y} \leftarrow rand(K,N)$

2. // Calcul de la décomposition en matrices non négatives de la matrice d'adjacence **A** :

   répéter

   pour $k=1\ldots K$ et $j=1\ldots N$ **faire**

   $$y_{kj} \leftarrow y_{kj} \frac{\left(\mathbf{X}^T\mathbf{A}\right)_{kj}}{\left(\mathbf{X}^T\mathbf{X}\mathbf{Y}\right)_{kj}+10^{-9}}$$

   fin

   pour $i=1\ldots N$ et $k=1\ldots K$ **faire**

   $$x_{ik} \leftarrow x_{ik} \frac{\left(\mathbf{A}\mathbf{Y}^T\right)_{ik}}{\left(\mathbf{X}\mathbf{Y}\mathbf{Y}^T\right)_{ik}+10^{-9}}$$

   fin

   jusqu'à *convergence*

3. // Normalisation des matrices **X** et **Y** et calcul du coefficient λ de chaque composante :

   pour $k=1\ldots K$ **faire**

   $\lambda_k = \|\mathbf{x}_{.k}\|_2 \times \|\mathbf{y}_{k.}\|_2$ // $\|\mathbf{v}\|_2 = \sqrt{\sum_{i=1} v_i^{\,2}}$ est la norme L2 du vecteur **v**

   $$\mathbf{x}_{.k} = \frac{\mathbf{x}_{.k}}{\|\mathbf{x}_{.k}\|_1}$$

   $$\mathbf{y}_{k.} = \frac{\mathbf{y}_{k.}}{\|\mathbf{y}_{k.}\|_1}$$

   fin

4. // Calcul du vecteur d'autorité **o** et du vecteur d'hubité **h** :

   $\mathbf{o} \leftarrow \mathbf{0}_{N\times 1}$; $\mathbf{h} \leftarrow \mathbf{0}_{N\times 1}$ // $\mathbf{0}_{N\times 1}$ est un vecteur colonne de dimension n contenant des zéros

   pour $k=1\ldots K$ **faire**

   $\mathbf{o} \leftarrow \mathbf{o} + \lambda_k \mathbf{y}_{k.}^{\,T}$

   $\mathbf{h} \leftarrow \mathbf{h} + \lambda_k \mathbf{x}_{.k}$

   fin

5. // Normalisation des vecteurs **o** et **h** :

   $$\mathbf{o} \leftarrow \frac{\mathbf{o}}{\|\mathbf{o}\|_1} \;;\; \mathbf{h} \leftarrow \frac{\mathbf{h}}{\|\mathbf{h}\|_1}$$ // $\|\mathbf{v}\|_1 = \sum_{i=1} v_i$ est la norme L1 du vecteur **v**

fin

2.3.5 Effet du nombre de dimensions *K* et convergence de l'algorithme

Le paramètre *K* de l'algorithme NHITS a un impact important sur les vecteurs d'autorité et d'hubité qu'il calcule. Si le graphe à analyser contient par exemple *C* communautés, une valeur de *K* inférieure à *C* fera que l'algorithme ignorera *C-K* communautés secondaires. Une valeur de *K* supérieure à *C* calculera en plus des vecteurs d'autorité (resp. d'hubité) qui ne correspondent pas à des communautés. En pratique, le nombre exact de communautés n'est pas connu et trouver ce nombre est une problématique difficile comme on le verra dans la seconde partie de cette thèse.

Un autre inconvénient de l'algorithme NHITS est lié à sa convergence. En fait, comme il est basé sur la décomposition en matrices non-négatives (NMF) qui est elle-même un algorithme local, la qualité de la solution finale (i.e. des vecteurs d'autorité et d'hubité) calculée par NHITS dépend de l'étape l'initialisation. Une méthode souvent utilisée pour choisir la meilleure décomposition en matrices non-négatives consiste à lancer l'algorithme NMF avec différentes initialisations, puis de garder la meilleure décomposition en termes de distance euclidienne par rapport à la matrice décomposée. C'est cette méthode que nous avons utilisée avec l'algorithme NHITS pour s'assurer de l'obtention de "bons" vecteurs d'autorité et d'hubité.

2.4 L'algorithme DocRank

2.4.1 Principe

Lors de nos expérimentations avec différents algorithmes d'analyse de liens existants, nous avons constaté que l'algorithme PageRank était le moins affecté par l'effet TKC. Cette observation a notamment été reportée dans [Tsaparas 03]. En cherchant la raison de cette robustesse de PageRank, il nous est apparu que celle-ci est due à la normalisation qu'il effectue avant de calculer le vecteur de centralité. En effet, avant d'appliquer la méthode des puissances avec la matrice d'adjacence, PageRank commence d'abord par normaliser cette matrice de telle sorte que la somme de chacune de ses lignes (i.e. somme des liens sortants de chaque document) soit égale à un. Rappelons également que PageRank est un algorithme itératif où à chaque itération, chaque document transfère aux documents qu'il cite une quantité égale à son PageRank divisé par le nombre de ses liens sortants ; cela permet de s'assurer que les documents qu'il cite reçoivent la même quantité.

Le troisième algorithme que nous proposons à savoir DocRank a pour but de pallier au problème de l'effet TKC en s'appuyant sur l'idée de la *normalisation*. DocRank ressemble à la fois à l'algorithme HITS et à l'algorithme PageRank. Il ressemble à HITS car il calcule un degré d'autorité et un degré d'hubité pour chaque nœud, et il ressemble à PageRank car il est basé sur le principe des marches aléatoires.

L'algorithme DocRank consiste à calculer la distribution stationnaire des deux chaînes de Markov suivantes: la chaîne *MCo* permettant d'obtenir des degrés d'autorité et la chaîne *MCh* permettant d'obtenir des degrés d'hubité. Les matrices de transition $\mathbf{M}_o$ et $\mathbf{M}_h$ respectivement des chaînes *MCo* et *MCh* sont définies par :

$$\left(\mathbf{M}_o\right)_{ij} = \frac{\left(\left(\mathbf{LA}\right)^T\left(\mathbf{LA}\right)\right)_{ij}}{\sum_{k=1}^{N}\left(\left(\mathbf{LA}\right)^T\left(\mathbf{LA}\right)\right)_{ik}} \tag{2.5}$$

$$\left(\mathbf{M}_h\right)_{ij} = \frac{\left(\left(\mathbf{AC}\right)^T\left(\mathbf{AC}\right)\right)_{ij}}{\sum_{k=1}^{N}\left(\left(\mathbf{AC}\right)^T\left(\mathbf{AC}\right)\right)_{ik}} \tag{2.6}$$

où : - $\mathbf{A}$ est la matrice d'adjacence.

- $\mathbf{L}$ est une matrice diagonale de dimension N telle que $l_{ii} = \left(d_i^{out}\right)^{-\alpha} = \left(\sum_{j=1}^{N} a_{ij}\right)^{-\alpha}$.
- $\mathbf{C}$ est une matrice diagonale de dimension N telle que $c_{ii} = \left(d_i^{in}\right)^{-\alpha} = \left(\sum_{j=1}^{N} a_{ji}\right)^{-\alpha}$.
- $\alpha \in \mathbb{R}$ est un paramètre de normalisation.

La matrice $\mathbf{M}_o$ (resp. $\mathbf{M}_h$) correspond en fait à une version (doublement) normalisée de la matrice d'autorité $\mathbf{A}^T\mathbf{A}$ (resp. d'hubité $\mathbf{AA}^T$) utilisée par HITS. Le calcul de $\mathbf{M}_o$ (resp. $\mathbf{M}_h$) revient à calculer dans un premier temps une matrice d'autorité (resp. d'hubité) non pas à partir de la matrice d'adjacence $\mathbf{A}$ mais plutôt à partir d'une matrice d'adjacence dont les lignes (resp. les colonnes) ont été normalisées (i.e. la matrice $\mathbf{LA}$ (resp. $\mathbf{AC}$)), puis dans un deuxième temps à normaliser cette "nouvelle" matrice d'autorité (resp. d'hubité) de telle sorte que la somme de ses lignes soit égale à un (i.e. pour la rendre stochastique).

2.4.2 Distributions stationnaires

Nous allons maintenant nous intéresser au calcul de la distribution stationnaire de la chaîne de Markov *MCo* (resp. *MCh*). Nous allons considérer d'abord le cas où les deux chaînes de Markov sont irréductibles. La matrice $\mathbf{M}_o$ (resp. $\mathbf{M}_h$) étant également apériodique implique que la chaîne de Markov *MCo* (resp. *MCh*) possède une distribution stationnaire unique. Cette distribution peut être calculée en utilisant une méthode itérative telle que la méthode des puissances itérées. Cependant, nous montrons ci-dessous qu'il existe en fait une forme close à cette distribution.

Théorème 1 :

Si la chaîne de Markov *MCo* est irréductible, alors elle possède une distribution stationnaire unique $\mathbf{o} = \left(o_1, \ldots, o_N\right)$ donnée par :

$$o_j = \frac{\sum_{i=1}^{N} a_{ij}\left(d_i^{out}\right)^{-2\alpha+1}}{\sum_{i=1}^{N}\left(d_i^{out}\right)^{-2\alpha+2}} \tag{2.7}$$

où $\mathbf{A} = \left(a_{ij}\right)_{\substack{i=1\ldots N \\ j=1\ldots N}}$ est la matrice d'adjacence et $d_i^{out} = \sum_{k=1}^{N} a_{ik}$ est le nombre de liens sortants du nœud *i*.

Preuve :

Nous allons montrer que la distribution $\mathbf{o}$ vérifie la condition $\mathbf{o} = \mathbf{M}_o{}^T \mathbf{o}$ (i.e. pour tout $j = 1...N$ $\mathbf{o}_j = \left(\mathbf{M}_o{}^T \mathbf{o}\right)_j$).

La matrice $\mathbf{M}_o$ peut être réécrite de la manière suivante :

$$\begin{aligned}
\left(\mathbf{M}_o\right)_{mj} &= \frac{\left(\left(\mathbf{LA}\right)^T \left(\mathbf{LA}\right)\right)_{mj}}{\sum_{k=1}^{N}\left(\left(\mathbf{LA}\right)^T \left(\mathbf{LA}\right)\right)_{mk}} \\
&= \frac{\sum_{i=1}^{N}\left(\mathbf{LA}\right)_{im}\left(\mathbf{LA}\right)_{ij}}{\sum_{k=1}^{N}\sum_{i=1}^{N}\left(\mathbf{LA}\right)_{im}\left(\mathbf{LA}\right)_{ik}} \\
&= \frac{\sum_{i=1}^{N}\left(a_{im} \times \left(d_i^{out}\right)^{-\alpha}\right)\left(a_{ij} \times \left(d_i^{out}\right)^{-\alpha}\right)}{\sum_{k=1}^{N}\sum_{i=1}^{N}\left(a_{im} \times \left(d_i^{out}\right)^{-\alpha}\right)\left(a_{ik} \times \left(d_i^{out}\right)^{-\alpha}\right)} \\
&= \frac{\sum_{i=1}^{N}\left(a_{im} \times a_{ij} \times \left(d_i^{out}\right)^{-2\alpha}\right)}{\sum_{i=1}^{N}\left(a_{im} \times \left(d_i^{out}\right)^{-2\alpha} \times \left(\sum_{k=1}^{N} a_{ik}\right)\right)} \\
&= \frac{\sum_{i=1}^{N}\left(a_{im} \times a_{ij} \times \left(d_i^{out}\right)^{-2\alpha}\right)}{\sum_{i=1}^{N}\left(a_{im} \times \left(d_i^{out}\right)^{-2\alpha+1}\right)}
\end{aligned}$$

Nous avons ainsi :

$$\begin{aligned}
\left(\mathbf{M}_o{}^T \mathbf{o}\right)_j &= \sum_{m=1}^{N}\left(\mathbf{M}_o{}^T\right)_{jm} o_m \\
&= \sum_{m=1}^{N}\left(\mathbf{M}_o\right)_{mj} o_m \\
&= \sum_{m=1}^{N}\left[\frac{\sum_{i=1}^{N}\left(a_{im} \times a_{ij} \times \left(d_i^{out}\right)^{-2\alpha}\right)}{\sum_{i=1}^{N}\left(a_{im} \times \left(d_i^{out}\right)^{-2\alpha+1}\right)} \frac{\sum_{i=1}^{N} a_{im}\left(d_i^{out}\right)^{-2\alpha+1}}{\sum_{i=1}^{N}\left(d_i^{out}\right)^{-2\alpha+2}}\right] \\
&= \sum_{m=1}^{N} \frac{\sum_{i=1}^{N}\left(a_{im} \times a_{ij} \times \left(d_i^{out}\right)^{-2\alpha}\right)}{\sum_{i=1}^{N}\left(d_i^{out}\right)^{-2\alpha+2}} \\
&= \frac{\sum_{i=1}^{N} a_{ij} \times \left(d_i^{out}\right)^{-2\alpha}\left(\sum_{m=1}^{N} a_{im}\right)}{\sum_{i=1}^{N}\left(d_i^{out}\right)^{-2\alpha+2}} \\
&= \frac{\sum_{i=1}^{N} a_{ij} \times \left(d_i^{out}\right)^{-2\alpha+1}}{\sum_{i=1}^{N}\left(d_i^{out}\right)^{-2\alpha+2}} \\
&= o_j \qquad \square
\end{aligned}$$

Théorème 2 :

Si la chaîne de Markov MCh est irréductible, alors elle possède une distribution stationnaire unique $\mathbf{h} = (h_1, \ldots, h_N)$ donnée par :

$$h_j = \frac{\sum_{i=1}^{N} a_{ji} \left(d_i^{in}\right)^{-2\alpha+1}}{\sum_{i=1}^{N} \left(d_i^{in}\right)^{-2\alpha+2}} \tag{2.8}$$

où $\mathbf{A} = \left(a_{ij}\right)_{\substack{i=1\ldots N \\ j=1\ldots N}}$ est la matrice d'adjacence et $d_i^{in} = \sum_{k=1}^{N} a_{ki}$ est le nombre de liens sortants du nœud *i*.

Preuve :

Un raisonnement similaire à celui utilisé pour prouver le théorème 1 peut être utilisé pour montrer que $\mathbf{h}$ vérifie la condition $\mathbf{h} = \mathbf{M}_h{}^T \mathbf{h}$.

Ces deux théorèmes permettent d'avoir une interprétation intuitive de la centralité calculée par l'algorithme DocRank. Ils montrent en effet que le degré d'autorité (resp. d'hubité) calculé par DocRank pour un nœud *i* correspond en quelque sorte au degré entrant (resp. sortant) normalisé de ce nœud.

Dans l'algorithme DocRank, chaque lien peut être considéré comme étant une recommandation. Le paramètre α sert à pondérer les différentes recommandations afin de diminuer l'effet TKC. Le degré d'autorité (resp. d'hubité) d'un nœud est alors égal à la somme des poids des recommandations qu'il reçoit (resp. qu'il effectue).

Le paramètre α joue un rôle crucial dans les performances de l'algorithme DocRank. Suivant la valeur qu'il prend, on a les différents cas suivants :

- $\alpha = 0.5$: le poids des recommandations est indépendant du degré du nœud et il est égal à un. L'algorithme dans ce cas est équivalent à la centralité de degré.

- $\alpha < 0.5$: les recommandations ont un poids proportionnel au degré du nœud. Cela revient à favoriser les recommandations émanant de (ou reçues par des) nœuds ayant un fort degré. En d'autres termes, une telle valeur aura pour conséquence de favoriser l'effet TKC.

- $\alpha > 0.5$: les recommandations ont un poids inversement proportionnel au degré du nœud. Cela revient à pénaliser les recommandations émanant de (ou reçues par des) nœuds ayant un fort degré. En d'autres termes, une telle valeur aura pour conséquence la réduction de l'effet TKC.

Un autre cas particulièrement intéressant est celui qui correspond à un DocRank avec $\alpha = 1$. Le poids de chaque recommandation est alors égal à l'inverse du degré du nœud. En d'autres termes, la somme des poids des recommandations d'un nœud est égale à 1. Cela revient à supposer que tous les nœuds possèdent le même "force" de recommandation et que ceux-ci sont tous égaux. Il s'agit en fait d'une version "démocratique" du modèle DocRank.

Dans le cas où la chaîne de Markov Mo (resp. Mh) n'est pas irréductible, nous montrons qu'il est toujours possible d'utiliser le résultat du théorème 1 (resp. théorème 2).

2.4.3 Détails de l'algorithme

Les détails de l'algorithme DocRank sont donnés par l'algorithme 2.3.

2.4.4 Exemples jouets

Les vecteurs d'autorité obtenus en utilisant DocRank avec $\propto$=1 sur les graphes jouets *G1* et *G3* sont respectivement indiqués par les tableaux 2.10 et 2.11. Pour le graphe *G1*, DocRank attribue 51% de l'autorité à la communauté {4,5,6} et 49% à la communauté {10,11,12}. Pour le graphe *G3*, le même algorithme attribue 35.5% de l'autorité à la communauté {4,5,6}, 30% à la communauté {10, 11} et 34.5% à la communauté {16,17,18}. Nous remarquons ainsi que la normalisation effectuée par DocRank lui permet de distribuer l'autorité (et l'hubité) aux nœuds appartenant à différentes communautés. Le paramètre $\propto$ de l'algorithme permet de contrôler la façon avec laquelle cette distribution doit être faite.

Algorithme 2.3 : L'algorithme DocRank

Entrées : - un graphe $G=(V,E)$ d'ordre *N* représenté par sa matrice d'adjacence **A**
- $\propto$: paramètre de normalisation

Sortie : vecteurs d'autorité **o** et d'hubité **h**

début

1. // Calcul des matrices diagonales de normalisation $\mathbf{L}$ et $\mathbf{C}$:

$\mathbf{L} \leftarrow diag\left(\mathbf{A} \times \mathbf{1}_{N\times 1}\right)$

$\mathbf{C} \leftarrow diag\left(\mathbf{A}^T \times \mathbf{1}_{N\times 1}\right)$ // $\mathbf{1}$ est un vecteur colonne de dimension *N* contenant des uns

pour $i = 1 \ldots N$ **faire**

  si $l_{ii} \neq 0$ **alors** $l_{ii} \leftarrow l_{ii}^{-\alpha}$

  si $c_{ii} \neq 0$ **alors** $c_{ii} \leftarrow c_{ii}^{-\alpha}$

fin

2. // Calcul des matrices d'adjacence normalisées $\mathbf{A}_o$ et $\mathbf{A}_h$:

$\mathbf{A}_o \leftarrow \mathbf{LA}$

$\mathbf{A}_h \leftarrow \mathbf{AC}$

3. // Calcul du vecteur d'autorité $\mathbf{o}$ et du vecteur d'hubité $\mathbf{h}$:

$\mathbf{o} \leftarrow \mathbf{A}_o^T \times \mathbf{1}_{N\times 1}$

$\mathbf{h} \leftarrow \mathbf{A}_h \times \mathbf{1}_{N\times 1}$

4. // Normalisation des vecteurs $\mathbf{o}$ et $\mathbf{h}$:

$\mathbf{o} \leftarrow \frac{\mathbf{o}}{\|\mathbf{o}\|_1}$

$\mathbf{h} \leftarrow \frac{\mathbf{h}}{\|\mathbf{h}\|_1}$ // $\|\mathbf{x}\|_1 = \sum_{i=1} x_i$ est la norme L1 du vecteur $\mathbf{x}$

fin

Tableau 2.10 - Degrés d'autorité et calculés par DocRank ($\propto=1$) pour les nœuds du graphe *G1*

Nœud	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*
Degré d'autorité	0	0	0	0.17	0.17	0.17	0	0	0	0.19	0.11	0.19

Tableau 2.11 - Degrés d'autorité calculés par DocRank ($\propto=1$) pour les nœuds du graphe *G3*

Nœud(s)	1,2,3,7,8,9,12,13,14,15	4	5	6	10	11	16	17	18
Degré d'autorité	0	0.10	0.10	0.15	0.15	0.15	0.17	0.06	0.11

2.5 Environnement d'expérimentation

Dans cette section, nous présentons la méthodologie utilisée pour l'évaluation des différents algorithmes de calcul de centralité proposés dans le cadre de cette thèse. Nous décrivons notamment les données ainsi que les mesures utilisées pour la comparaison des différents algorithmes.

2.5.1 Graphes de documents utilisés

Les tableaux 2.12 et 2.13 résument les propriétés des huit graphes de documents que nous avons utilisés pour nos différentes expérimentations. Ces huit graphes sont de nature différente puisque quatre d'entre eux sont des graphes de pages web alors que les quatre autres sont des graphes d'articles scientifiques.

Nous avons construit les graphes des tableaux 2.12 et 2.13 en utilisant la méthode proposée par Kleinberg dans [Kleinberg 99a]. Celle-ci consiste à interroger un moteur de recherche avec une requête afin de récupérer les 200 premiers documents retournés par le moteur. Cet ensemble initial de pages web appelé "Root Set (RS)" est ensuite étendu en deux étapes : dans la première étape (extension avant), un crawler est utilisé afin de rajouter à RS les documents pointés par un document appartenant à RS (en limitant ce nombre à 50 par document) ; la deuxième étape (extension arrière) consiste à rajouter au "root set" des documents qui pointent vers des pages appartenant à RS (en limitant également ce nombre à 50 par document). La deuxième étape nécessite l'utilisation d'un moteur de recherche afin d'avoir la liste des URL qui pointent vers une page web donnée.

Les graphes Apple, Armstrong, Jaguar et Washington ont ainsi été construits en interrogeant le moteur de recherche Google avec des requêtes qui correspondent aux noms des graphes. Les graphes Plasma_Physics et Solar_Wind ont quant à eux été collectés en interrogeant la base documentaire de l'ADS en utilisant respectivement les requêtes "Plasma physics" et "Solar wind".

Enfin, les graphes Citeseer et Cora sont des graphes d'articles scientifiques dans le domaine de l'informatique. Ils ont été collectés et annotés par Lu et Getoor [Lu and Getoor 03] ; ils sont disponibles à l'adresse [URL 1]. Chaque document de Cora appartient à l'une des thématiques suivantes : réseaux de neurones, algorithmes génétiques, apprentissage par renforcement, théorie de l'apprentissage, apprentissage de règles, méthodes probabilistes, et

raisonnement par cas. Les documents de Citeseer appartiennent à cinq thématiques : agents, bases de données, recherche d'information, apprentissage, et interaction homme-machine.

Propriété	**Apple**	**Armstrong**	**Jaguar**	**Washington**
Nombre de documents	1894	3270	2233	4912
Nombre de liens	36124	4695	30094	13776
Nombre moyen de liens par document	19.07	1.44	13.48	2.80
Nombre de documents ayant des liens entrants	900	604	1011	1589
Nombre de documents ayant des liens sortants	1680	2887	1928	4191
Nombre de composantes dans le graphe G_o (ou G_h)	5	40	14	17
Taille de la composante la plus grande dans G_o	895	408	963	1498
Taille de la composante la plus grande dans G_h	1676	2680	1894	4167
Type de documents	Pages web	Pages web	Pages web	Pages web
Nombre de thématiques	-	-	-	-

Tableau 2.13 – Propriétés des graphes de pages web utilisés pour les expérimentations

Propriété	**Citeseer**	**Cora**	**Plasma_Physics**	**Solar_Wind**
Nombre de documents	2994	2708	5151	4262
Nombre de liens	4277	5429	16635	35248
Nombre moyen de liens par document	1.43	2	3.23	8.27
Nombre de documents ayant des liens entrants	1760	1565	2561	2944
Nombre de documents ayant des liens sortants	2099	2222	4794	3971
Nombre de composantes dans le graphe G_o (ou G_h)	570	162	2	1
Taille de la composante la plus grande dans G_o	847	1330	2559	2944
Taille de la composante la plus grande dans G_h	1123	1961	4793	3971
Type de documents	Articles	Articles	Articles	Articles
Nombre de thématiques	5	7	-	-

Tableau 2.14 - Propriétés des graphes d'articles scientifiques utilisés pour les expérimentations

2.5.2 Mesures d'évaluation

En recherche d'information, l'évaluation des algorithmes de calcul de centralité (appelés aussi algorithmes de "ranking") se fait généralement en utilisant une collection de documents où chaque document a été classé comme pertinent ou pas pertinent par des experts [Baeza-Yates and Ribeiro-Neto 99][Manning et al. 08]. La liste des *K* documents retournée par l'algorithme de "ranking" est alors évaluée à l'aide de mesures classiques telles que la précision et le rappel. Cependant, pour l'évaluation de l'effet TKC, cette méthode n'est pas adaptée car il se pourrait que les *K* premiers documents soient pertinents alors qu'ils ne traitent qu'un seul aspect de la requête (i.e. ils appartiennent tous à la même communauté) sachant que d'autres aspects existent aussi (i.e. d'autres communautés).

L'effet TKC ayant reçu peu d'attention de la part des chercheurs, nous n'avons pu trouver dans la littérature des critères quantitatifs permettant de le mesurer. Le peu de travaux ayant traité cette problématique notamment [Borodin et al. 01] et [Lempel and Moran 00] se contentent d'évaluations subjectives où la liste des *K* documents ayant le plus fort degré d'autorité (ou d'hubité) est présentée et commentée. C'est pourquoi nous avons jugé utile de proposer deux mesures complémentaires permettant de comparer la robustesse de plusieurs algorithmes à l'effet TKC.

La première mesure que nous proposons attribue un score à chaque algorithme de "ranking" en se basant sur l'intuition qu'un "bon" algorithme de calcul de centralité dans les graphes de documents qui simultanément:

- assigne un fort degré d'autorité (resp. d'hubité) aux documents ayant un grand nombre de liens entrants (resp. sortants). Cette condition est motivée par le fait que de nombreuses études ont montré qu'il y avait une forte corrélation entre l'importance d'un document et le degré de celui-ci [Fortunato 08].

- retourne une liste contenant des documents appartenant à différentes communautés (ou différentes thématiques). Cette idée sera capturée par la faible similarité entre les documents calculée en utilisant les liens uniquement.

Plus précisément, le premier critère que nous proposons correspond au rapport entre la somme des degrés des *K* documents les plus importants (selon l'algorithme de calcul de centralité) et la somme des similarités entre ces mêmes documents. Pour l'évaluation de la liste des *K* documents ayant le plus fort degré d'autorité, nous proposons ainsi l'indice ORQ (Authorities Ranking Quality) :

$$ORQ(\mathbf{x}) = \frac{\sum_{i=1}^{K} d^{in}(x_i)}{1+\sum_{i=1}^{K}\sum_{j=i+1}^{K} co(x_i, x_j)} \tag{2.9}$$

où $\mathbf{x}$ est un vecteur contenant les numéros des *K* documents classés comme meilleurs autorités, $d^{in}(x_i)$ est le degré entrant du document x_i et $co(x_i, x_j)$ est la similarité de co-citation [Small 73] entre les documents x_i et x_j. Plus la valeur de cet indice est grande meilleure est la qualité du "ranking".

De manière similaire, la liste des K documents ayant le plus fort degré d'hubité sera évaluée par l'indice HRQ (Hubs Ranking Quality) suivant :

$$HRQ(\mathbf{x}) = \frac{\sum_{i=1}^{K} d^{out}(x_i)}{1 + \sum_{i=1}^{K}\sum_{j=i+1}^{K} cb(x_i, x_j)} \tag{2.10}$$

où $\mathbf{x}$ est un vecteur de dimension K contenant les numéros des documents classés comme meilleurs hubs, $d^{out}(x_i)$ est le degré sortant du document x_i et $cb(x_i, x_j)$ est la similarité de couplage bibliographique [Kessler 63] entre les documents x_i et x_j. Plus la valeur de cet indice est grande, meilleure est la qualité du "ranking".

La deuxième mesure que nous proposons permet de mesurer la diversité thématique au sein de la liste des K documents les plus importants. Cette mesure nécessite toutefois de disposer d'informations concernant la thématique de chaque document (ces informations sont aussi dites externes). Pour l'évaluation de la diversité thématique au sein de la liste des K documents ayant le plus fort degré d'autorité, nous proposons l'indice TDO (Topic Diversity of Authorities) défini par :

$$TDO(\mathbf{c}) = -\sum_{i=1}^{T} \frac{c_i}{K} \log\left(\frac{c_i}{K}\right) \tag{2.11}$$

où $\mathbf{c}$ est un vecteur de dimension T tel que chaque entrée c_i de ce vecteur indique le nombre de documents (parmi K) appartenant à la thématique numéro i. Il s'agit en fait d'une entropie qui vaut 0 si les K documents appartiennent à la même thématique. Plus la valeur de cet indice est grande plus il y a de thématiques dans la liste des K meilleures autorités.

Pour l'évaluation de la diversité thématique dans la liste des K documents ayant le plus fort degré d'hubité, nous proposons l'indice TDH (Topic Diversity of Hubs) défini par :

$$TDH(\mathbf{c}) = -\sum_{i=1}^{T} \frac{c_i}{K} \log\left(\frac{c_i}{K}\right) \tag{2.12}$$

où $\mathbf{c}$ est un vecteur de dimension T tel que chaque entrée c_i de ce vecteur indique le nombre de documents hubs (parmi K) appartenant à la thématique numéro i. Il s'agit en fait d'une entropie qui vaut 0 si les K documents appartiennent à la même thématique. Plus la valeur de cet indice est grande plus il y a de thématiques dans la liste des K meilleurs hubs.

Les indices *ORQ* et *HRQ* permettent de mesurer l'effet TKC sans utiliser d'informations externes tandis que les critères *TDO* et *TDH* permettent de mesurer la diversité thématique en utilisant des informations additionnelles concernant la thématique de chaque document.

2.5.3 Algorithmes comparés :

Lors de nos expérimentations, nous avons comparé les onze algorithmes suivants :

DEG (centralité de degré, équivalente à DocRank avec $\alpha = 0.5$) – HITS – PageRank – SALSA – HubAvg – MHITS – NHITS_10 (NHITS avec $K = 10$) – NHITS_20 ($K = 20$) – DocRank_0 (DocRank avec $\alpha = 0$) – DocRank_1 ($\alpha = 1$) – DocRank_15 ($\alpha = 1.5$).

2.6 Résultats expérimentaux

2.6.1 Evaluation de la qualité du classement

Pour chacun des algorithmes étudiés et pour chaque graphe de documents, nous donnons la liste des dix documents ayant le plus fort degré d'autorité, ainsi que la liste des dix documents ayant le plus fort degré d'hubité (voir tableaux 2.15 à 2.28). Nous reportons également pour chaque algorithme, les valeurs des mesures ORQ@10 (ORQ avec K=10), ORQ@20, HRQ@10 et HRQ@20. Cependant, le nombre de résultats étant important, nous présentons dans cette section uniquement les listes des 10 documents ayant le plus fort degré d'autorité retournées par HITS, PageRank et DocRank_1. Les listes (des meilleurs autorités et des meilleurs hubs) retournées par les autres algorithmes sont reportées dans l'annexe B.

Sur le plan qualitatif, nous constatons que :

- HITS se focalise toujours sur une seule communauté (ou thématique). Cette thématique est souvent peu pertinente ou très générale par rapport à la requête qui a permis de construire le graphe de documents ; on parle alors de "topic drift" ou de "topic generalization". C'est le cas par exemple des résultats obtenus avec le graphe *Armstrong* où les documents ayant les plus forts degrés d'autorité sont des sites de sport (Nhl, Nba, etc.). La liste des dix meilleurs hubs retournée par HITS nous permet d'avoir une explication pour ces résultats : plusieurs de ces hubs parlent d'un joueur de hockey professionnel qui s'appelle *Derek Armstrong*.

- PageRank donne de meilleurs résultats que l'algorithme HITS puisque des documents pertinents et traitant différentes thématiques sont présents dans la liste des dix meilleures autorités. Mais PageRank classe certains documents comme étant importants alors qu'ils ne sont pas pertinents par rapport à la requête initiale. Les résultats avec les graphes *Armstrong*, *Jaguar* et *Washington* illustrent bien cette situation où l'on retrouve dans la liste des meilleures autorités les sites du logiciel MediaWiki et de l'organisation WikiMediaFoundation.

- DocRank ne souffre pas de l'effet TKC puisque la liste des meilleures autorités qu'il calcule contient des documents pertinents et appartenant à différentes thématiques. Par exemple pour le graphe *Armstrong*, on retrouve une page sur le musicien Louis Armstrong, la page officielle du comté d'Armstrong aux Etats-Unis ou encore une page sur le cycliste Lance Armstrong.

D'un point de vue quantitatif, nous constatons que l'algorithme HITS obtient dans la plupart des cas les plus faibles valeurs d'ORQ et d'HRQ. La simple centralité de degré obtient de meilleurs résultats que HITS. PageRank obtient dans tous les cas des résultats largement meilleurs que ceux de HITS ; dans certains cas il est aussi meilleur que la centralité de degré. L'algorithme NHITS permet d'obtenir de meilleurs résultats que HITS mais ses performances restent toutefois bien en-dessous de celles de PageRank. DocRank quant à lui réalise les meilleures performances dans le sens où les documents qu'il classe comme étant importants ont à la fois beaucoup de liens et appartiennent à des communautés différentes. Cette dernière propriété lui permet de se distinguer de la centralité de degré qui ne tient compte que du nombre de liens.

Algorithme	Degré d'autorité	URL	Nombre de liens entrants	Nombre de liens sortants
HITS	0.0087516	http://www.xbox360fanboy.com	279	120
	0.0087142	http://www.secondlifeinsider.com	271	121
	0.0087108	http://www.cssinsider.com	251	47
	0.0086881	http://www.dsfanboy.com	257	121
	0.0086872	http://www.bbhub.com	253	116
	0.0086849	http://www.pspfanboy.com	270	116
	0.008683	http://www.ps3fanboy.com	254	116
	0.0086821	http://www.nintendowiifanboy.com	254	121
	0.0086768	http://www.wowinsider.com	252	116
	0.0086758	http://www.droxy.com	251	121
PageRank	0.029706	http://www.download.com/itunes-for-windows/...	46	12
	0.015402	http://www.versiontracker.com	71	1
	0.014821	http://www.macfixit.com	76	1
	0.013073	http://www.apple.com	280	0
	0.011134	http://store.apple.com	47	0
	0.010116	http://www.macworld.com	122	7
	0.0092659	http://www.mac.com	57	1
	0.0074745	http://www.imaconlinepoker.com	3	1
	0.0071086	http://www.macrumors.com In:98 Out:10	98	10
	0.0071011	http://www.macpokeronline.com In:9 Out:1	9	1
DocRank1	0.041252	http://www.apple.com	280	0
	0.017196	http://www.tuaw.com	351	157
	0.013126	http://www.mac.com	57	1
	0.012662	http://www.engadget.com	319	130
	0.012131	http://www.apple-history.com	114	0
	0.011356	http://www.macworld.com	122	7
	0.011272	http://www.appleinsider.com	117	12
	0.011255	http://www.download.com/itunes-for-windows/...	46	12
	0.01118	http://www.macrumors.com	98	10
	0.010859	http://developer.apple.com/wwdc In:84 Out:0	84	0

Tableau 2.15 – Liste des 10 documents ayant le plus fort degré d'autorité dans le graphe Apple

Algorithme	ORQ@10	ORQ@20	HRQ@10	HRQ@20
DEG	9.0591	1.7336	3.5591	0.8070
HITS	5.8095	1.3598	2.7746	0.7426
PageRank	16.1786	6.3757	-	-
Salsa	9.0591	1.7336	3.5591	0.8070
HubAvg	7.1820	1.5430	0.2074	0.0584
MHITS	5.8095	1.3598	2.7746	0.7426
NHITS_10	7.0691	1.5287	3.0895	0.7126
NHITS_20	7.1820	1.5497	2.9585	0.6905
DocRank_0	5.8095	1.3598	2.7746	0.7366
DocRank_1	26.0670	4.9819	**15.5946**	4.6577
DocRank_1.5	**29.3962**	**7.5850**	**15.5946**	**5.4337**

Tableau 2.16 – Qualité de classement avec le graphe Apple

Algorithme	Degré d'autorité	URL	Nombre de liens entrants	Nombre de liens sortants
HITS	0.02958	http://www.nhl.com	11	0
	0.029067	http://www.nba.com	8	0
	0.02891	http://www.golfdigest.com	7	0
	0.028754	http://www.nbcolympics.com/index.html?qs=pt=espn	6	0
	0.028754	http://espnsportsfigures.com	6	0
	0.028754	http://www.espnbooks.com	6	0
	0.028754	http://espnzone.com	6	0
	0.028754	http://joinourteam.espn.com/joinourteam	6	0
	0.028754	http://www.wnba.com	6	0
	0.028754	http://www.jayski.com	6	0
PageRank	0.024714	http://www.mediawiki.org	56	1
	0.023196	http://wikimediafoundation.org	49	1
	0.022289	http://www.livestrong.org	64	2
	0.016661	http://www.rdale.k12.mn.us/ahs	51	1
	0.014649	http://www.store-laf.org	10	1
	0.014279	http://www.schoolidentity.com/schoolid/store.html...	1	1
	0.01018	http://www.livestrongarmy.org	3	2
	0.0095567	http://www.sanbornwebdesigns.com	3	1
	0.0095567	http://www.glassart.biz	3	1
	0.0078046	http://www.thepaceline.com	56	8
DocRank1	0.019575	http://www.satchmo.net	100	4
	0.017716	http://www.armstrongcounty.com	58	2
	0.017417	http://www.armstrong.com	63	3
	0.017002	http://www.armstronggarden.com	51	1
	0.016973	http://www.joearmstrong.com	50	1
	0.016886	http://www.armstrongmold.com	50	0
	0.016722	http://www.armstrongtools.com	51	0
	0.01651	http://www.livestrong.org	64	2
	0.016492	http://www.armstrongglass.com	52	4
	0.016488	http://www.livestrong.org/site/c.jvKZLbMRIsG/...	56	0

Tableau 2.17 – Liste des 10 documents ayant le plus fort degré d'autorité dans le graphe Armstrong

Algorithme	ORQ@10	ORQ@20	HRQ@10	HRQ@20
DEG	20.0403	7.7528	2.5950	1.0994
HITS	0.1625	0.0411	0.9735	0.2579
PageRank	8.5728	7.8711	-	-
Salsa	20.0403	7.7528	2.5950	1.0994
HubAvg	11.5987	6.3845	0.0217	0.0052
MHITS	0.1625	0.0411	0.9735	0.2579
NHITS_10	15.1622	5.5538	1.8800	0.6442
NHITS_20	21.4288	8.4479	1.8800	0.6225
DocRank_0	10.0496	0.6463	0.6273	0.3919
DocRank_1	**45.3566**	16.7184	6.4218	3.3901
DocRank_1.5	42.1440	**20.8215**	**16.7000**	**12.2148**

Tableau 2.18 – Qualité de classement avec le graphe Armstrong

Algorithme	Degré d'autorité	URL	Nombre de liens entrants	Nombre de liens sortants
HITS	0.0073099	http://www.cssinsider.com	170	78
	0.0072756	http://www.joystiq.com	175	121
	0.0072744	http://www.brianalvey.com	172	2
	0.007272	http://www.calacanis.com	171	4
	0.0072699	http://www.thediabetesblog.com	170	133
	0.0072694	http://www.wowinsider.com	170	133
	0.0072667	http://www.thecardioblog.com	169	133
	0.0072661	http://www.ps3fanboy.com	169	133
	0.0072661	http://www.pspfanboy.com	169	133
	0.0072655	http://www.cinematical.com	172	148
PageRank	0.019909	http://www.mediawiki.org	61	2
	0.015637	http://www.oreillylearning.com	36	4
	0.010475	http://www.jaguars.com	55	5
	0.0098033	http://wikimediafoundation.org	40	1
	0.009027	http://wikimediafoundation.org/wiki/privacy_policy	22	1
	0.0090211	http://www.flickr.com/photos/ableman/sets/72157...	46	46
	0.0085951	http://www.jag-lovers.org	146	4
	0.0072018	http://www.jagweb.com	72	4
	0.0056405	http://www.weblogsinc.com	139	67
	0.0053051	http://www.jcna.com	115	1
DocRank1	0.02085	http://www.jag-lovers.org	146	4
	0.018696	http://www.jaguar.com	119	0
	0.017324	http://www.flickr.com/photos/ableman/sets/72157...	46	46
	0.012429	http://www.jcna.com	115	1
	0.011683	http://www.apple.com/macosx	62	0
	0.010454	http://www.mediawiki.org	61	2
	0.010148	http://www.jaguars.com	55	5
	0.0094735	http://www.catdriver.com	81	0
	0.009247	http://www.apple.com	56	0
	0.0090213	http://www.jagbits.com	73	7

Tableau 2.19 – Liste des 10 documents ayant le plus fort degré d'autorité dans le graphe Jaguar

Algorithme	ORQ@10	ORQ@20	HRQ@10	HRQ@20
DEG	3.8350	0.9075	3.9694	0.9267
HITS	3.7597	0.8981	3.9694	0.9267
PageRank	16.8092	5.2635	-	-
Salsa	3.8350	0.9075	3.9694	0.9267
HubAvg	3.8218	0.9108	0.4429	0.4381
MHITS	3.7597	0.8981	3.9694	0.9267
NHITS_10	3.7819	0.9067	3.9658	0.9267
NHITS_20	3.7418	0.9000	3.9694	0.9267
DocRank_0	3.7471	0.8981	3.9694	0.9267
DocRank_1	18.9607	4.9154	**29.5837**	3.4580
DocRank_1.5	**24.7485**	**5.7807**	27.0480	**10.4353**

Tableau 2.20 – Qualité de classement avec le graphe Jaguar

Algorithme	Degré d'autorité	URL	Nombre de liens entrants	Nombre de liens sortants
HITS	0.027657	http://www.redskins.com	115	8
	0.025124	http://www.denverbroncos.com	36	1
	0.025102	http://www.sf49ers.com	36	0
	0.025059	http://www.chicagobears.com	36	1
	0.025035	http://www.neworleanssaints.com	35	1
	0.025035	http://www.stlouisrams.com	35	0
	0.025034	http://www.seahawks.com	50	0
	0.024962	http://www.steelers.com	39	0
	0.024742	http://www.buffalobills.com	35	0
	0.024724	http://www.chargers.com	36	0
PageRank	0.019436	http://www.mediawiki.org	102	2
	0.010539	http://www.usa.gov	56	0
	0.010256	http://www.parks.wa.gov	82	2
	0.0098836	http://wikimediafoundation.org	88	1
	0.009319	http://wikimediafoundation.org/wiki/Privacy_policy	65	1
	0.0078055	http://www.washingtonpost.com	203	38
	0.0062546	http://www.tvw.org/index.cfm	14	0
	0.0059141	http://www.lib.washington.edu	49	3
	0.0048118	http://www.washingtonwine.org	56	4
	0.0047809	http://www.k12.wa.us	73	4
DocRank1	0.013254	http://www.washingtonpost.com	203	38
	0.011511	http://access.wa.gov	188	0
	0.011337	http://www.washington.org	84	5
	0.010936	http://www.worldtimeserver.com/current_time_in...	48	8
	0.010817	http://geo.craigslist.org/iso/us/wa	48	1
	0.010374	http://www.imdb.com/name/nm0000243	48	1
	0.01029	http://www.flickr.com/photos/tags/washington	48	8
	0.010115	http://www.washingtoncountytn.com	50	3
	0.0099007	http://www.missingkids.com/precreate/WA.html	53	0
	0.009853	http://www.washingtonwine.org	56	4

Tableau 2.21 – Liste des 10 documents ayant le plus fort degré d'autorité dans le graphe Washington

Algorithme	ORQ@10	ORQ@20	HRQ@10	HRQ@20
DEG	25.8089	7.8042	8.0574	2.5515
HITS	1.1677	0.2324	1.1979	0.2481
PageRank	18.5486	9.3463	-	-
Salsa	25.8089	7.8042	8.0574	2.5515
HubAvg	23.1265	6.4972	0.0217	0.0065
MHITS	1.1677	0.2324	1.1979	0.2481
NHITS_10	6.9531	1.3479	5.5200	0.6624
NHITS_20	10.4264	3.3366	6.8020	0.9872
DocRank_0	3.4154	0.4221	3.5658	0.6172
DocRank_1	**59.1528**	29.3282	**17.6531**	7.8782
DocRank_1.5	48.6388	**30.8082**	**17.6531**	**11.0055**

Tableau 2.22 – Qualité de classement avec le graphe Washington

Algorithme	Degré d'autorité	Titre	Nombre de liens entrants	Nombre de liens sortants
HITS	0.067149	Dust-acoustic waves in dusty plasmas	143	7
	0.046614	Dusty plasmas in the solar system	107	26
	0.044212	Laboratory observation of the dust-acoustic wave mode	86	5
	0.034683	Plasma crystal: Coulomb … in a dusty plasma	100	4
	0.033738	Direct observation of Coulomb … dusty plasmas	97	0
	0.028541	Dust ion-acoustic wave	59	2
	0.026343	Cosmic Dusty Plasmas	70	58
	0.023738	The electrostatics of a dusty plasma	68	9
	0.021574	Laboratory studies of waves and … in dusty plasmas	53	34
	0.020679	Condensed Plasmas under Microgravity	66	0
PageRank	0.0047334	Centrifugally driven diffusion of Iogenic plasma	11	1
	0.0045094	Factors governing the ratio of inward to …	4	1
	0.0044475	Helical microtubules of graphitic carbon	37	0
	0.0043543	N-dependence in the classical one-component …	50	0
	0.0040362	A General Formula for the Estimation of Dielectro…	81	0
	0.0038163	Ionization Equilibrium and Radiative Cooling …	75	23
	0.0036043	Radiative cooling of a low-density plasma	69	16
	0.0035877	A survey of the plasma electron environment of Jupit...	14	3
	0.0033341	Strong turbulence of plasma waves	50	0
	0.0033016	A General Theory of the Plasma of an Arc	50	0
DocRank1	0.0092894	Plasma perspective on strong field multiphoton…	53	0
	0.0091347	Plasma Losses by Fast Electrons in Thin Films	50	0
	0.0090982	Interaction of "Solitons" in a Collisionless Plasma…	50	0
	0.0081908	The ULYSSES solar wind plasma experiment	51	0
	0.0079346	Fast-wave heating of a two-component plasma	50	0
	0.0078562	Plasma spectroscopy	64	0
	0.0076682	SWE, A Comprehensive Plasma Instrument for …	56	2
	0.0076624	Plasma source ion-implantation technique for …	50	7
	0.0076033	Random Phasing of High-Power Lasers for Uniform…	51	0
	0.0075752	CLOUDY 90: Numerical Simulation of Plasmas …	54	9

Tableau 2.23 – Liste des 10 documents ayant le plus fort degré d'autorité dans le graphe Plasma_Physics

Algorithme	ORQ@10	ORQ@20	HRQ@10	HRQ@20
DEG	19.4547	8.4364	18.5060	5.8929
HITS	7.5463	1.9671	1.0564	0.2449
PageRank	20.4058	16.1964	-	-
Salsa	19.4547	8.4364	18.5060	5.8929
HubAvg	7.3364	2.1075	0.1091	0.0320
MHITS	7.5463	1.9671	1.0564	0.2449
NHITS_10	9.3188	3.0671	3.5049	1.0759
NHITS_20	10.6941	4.2358	10.8649	2.3431
DocRank_0	13.3115	4.2760	2.6941	0.7480
DocRank_1	**50.0882**	34.1296	32.6085	10.7862
DocRank_1.5	50.0276	**34.1347**	**38.0190**	**19.6758**

Tableau 2.24 – Qualité de classement avec le graphe Plasma_Physics

Algorithme	Degré d'autorité	Titre	Nombre de liens entrants	Nombre de liens sortants
HITS	0.010016	Transition region, corona, and solar wind in coronal…	105	31
	0.0094714	UVCS/SOHO Empirical … in the Solar Corona	106	7
	0.0088182	Spectroscopic Constraints … Polar Solar Corona …	92	23
	0.0084365	Wave heating and acceleration … ions by cyclotron…	87	6
	0.0082958	Two-Fluid Model for Heating of the Solar Corona …	77	32
	0.0080378	Heating and cooling … cyclotron waves …	64	39
	0.0075321	On the preferential acceleration and heating of … ions	71	14
	0.0074118	Ion Cyclotron Wave Dissipation in the Solar Corona:…	63	32
	0.0074003	Transition region, corona, and solar wind in coronal…	72	13
	0.0073041	An Empirical Model of a Polar Coronal Hole at …	84	21
PageRank	0.012919	Plasma waves associated with energetic particles…	56	3
	0.0077485	Upstream particles observed in the earth's foreshock…	7	2
	0.0075253	Upstream particle spatial gradients and plasma waves	8	2
	0.0065994	Bi-directional streaming of solar wind electrons …	28	0
	0.0052161	Magnetic loop behind an interplanetary shock…	122	1
	0.0050967	Solar wind protons - Three-dimensional velocity…	109	2
	0.0050043	Dynamics of the Interplanetary Gas and Magnetic…	147	2
	0.0047329	Interplanetary dynamical processes.	125	0
	0.0043107	Microinstabilities upstream of the earth's bow shock -...	2	1
	0.0042364	Measurement of the rugged invariants of …	152	8
DocRank1	0.011246	Abundances of the elements - Meteoritic and solar	66	0
	0.0077556	The Angular Momentum of the Solar Wind	92	3
	0.0058943	Plasma waves associated with energetic particles …	56	3
	0.005427	Introduction to the solar wind	54	0
	0.005193	Solar Wind Electron Proton Alpha Monitor …	107	32
	0.0050014	Coronal Expansion and Solar Wind	106	0
	0.0048544	The ULYSSES solar wind plasma experiment	103	0
	0.0047523	Dynamics of the Interplanetary Gas and Magnetic …	147	2
	0.0047428	Energy coupling between the solar wind and the …	81	0
	0.0046232	Propagation of cosmic rays in the solar wind.	59	0

Tableau 2.25 – Liste des 10 documents ayant le plus fort degré d'autorité dans le graphe Solar_Wind

Algorithme	ORQ@10	ORQ@20	HRQ@10	HRQ@20
DEG	36.3293	8.3566	16.0667	3.9630
HITS	4.7387	1.2917	6.6282	1.1835
PageRank	24.3142	12.5359	-	-
Salsa	36.3293	8.3566	16.0667	3.9630
HubAvg	24.8750	6.7503	0.0217	0.0052
MHITS	4.7387	1.2917	6.6282	1.1835
NHITS_10	15.9486	3.8314	8.6744	1.8088
NHITS_20	14.0790	3.9032	11.0566	2.1481
DocRank_0	7.0564	2.8169	11.5372	2.2625
DocRank_1	42.1750	18.8009	**33.0878**	**8.5517**
DocRank_1.5	**56.1203**	**22.2334**	22.3814	8.3110

Tableau 2.26 – Qualité de classement avec le graphe Solar_Wind

Algorithme	ORQ@10	ORQ@20	HRQ@10	HRQ@20
DEG	8.2908	2.1329	2.1224	0.4706
HITS	2.9465	0.5591	0.8855	0.1584
PageRank	21.1597	6.3270	-	-
Salsa	8.2908	2.1329	2.1224	0.4706
HubAvg	2.9778	0.7696	0.0217	0.0097
MHITS	2.9465	0.5591	0.8855	0.1584
NHITS_10	4.8924	1.2638	0.8470	0.2742
NHITS_20	7.9427	1.5273	1.1298	0.3092
DocRank_0	3.1003	0.9437	0.8855	0.1706
DocRank_1	19.1740	4.8254	6.4474	2.4317
DocRank_1.5	**24.8579**	**11.8074**	**6.9545**	**4.4643**

Tableau 2.27 – Qualité de classement avec le graphe Citeseer

Algorithme	ORQ@10	ORQ@20	HRQ@10	HRQ@20
DEG	21.7665	7.6391	1.5625	0.7353
HITS	7.4769	2.9108	0.1839	0.0445
PageRank	14.4782	6.3116	-	-
Salsa	21.7665	7.6391	1.5625	0.7353
HubAvg	7.4228	3.4165	0.0217	0.0052
MHITS	7.4769	2.9108	0.1839	0.0445
NHITS_10	17.6386	4.8177	0.3327	0.1105
NHITS_20	20.4877	6.0414	0.3284	0.1454
DocRank_0	16.9253	7.4520	0.2296	0.0676
DocRank_1	27.9477	9.2448	**4.6000**	3.3571
DocRank_1.5	**28.3183**	**10.5355**	**4.6000**	**3.6667**

Tableau 2.28 – Qualité de classement avec le graphe Cora

2.6.2 Evaluation de la diversité thématique

Les tableaux 2.29 et 2.30 indiquent les résultats de la diversité thématique en utilisant les graphes Cora et Citeseer. Nous constatons clairement que l'algorithme HITS est celui qui souffre le plus de l'effet TKC et que l'algorithme DocRank avec $\alpha \geq 1$ réussit à faire ressortir des documents appartenant à diverses communautés (ou thématiques). Lorsque $\alpha < 0.5$ nous remarquons que les résultats obtenus sont conformes à l'interprétation du paramètre α que nous avons donnée dans la section 2.4.2 à savoir que DocRank favorise l'effet TKC.

Algorithme	**TDO@10**	**TDO@20**	**TDH@10**	**TDH@20**
DEG	0.8018	1.0331	0.6390	0.5182
HITS	0	0	0	0
PageRank	1.3138	1.4306	-	-
Salsa	0.8018	1.0331	0.6390	0.5182
HubAvg	0	0.3251	0.3251	0.1985
MHITS	0	0	0	0
NHITS_10	0.6390	0.6129	0	0.1985
NHITS_20	0.9503	0.9143	0.5004	0.5182
DocRank_0	0	0.3944	0	0
DocRank_1	1.4185	1.3443	**1.3592**	1.2610
DocRank_1.5	**1.4708**	**1.4582**	1.2206	**1.3452**

Tableau 2.29 – Diversité thématique avec le graphe Citeseer

Algorithme	**TDO@10**	**TDO@20**	**TDH@10**	**TDH@20**
DEG	1.5048	1.7490	1.2206	1.5401
HITS	0	0.1985	0	0
PageRank	1.2206	1.4150	-	-
Salsa	1.5048	1.7490	1.2206	1.5401
HubAvg	0	0.1985	0.3251	0.1985
MHITS	0	0.1985	0	0
NHITS_10	1.0297	1.4406	0.6109	1.0104
NHITS_20	1.2206	1.5954	1.0549	1.0671
DocRank_0	1.4185	1.6365	0.3251	0.3944
DocRank_1	1.6957	1.6782	1.6957	**1.7389**
DocRank_1.5	**1.8344**	**1.7127**	**1.8344**	1.5968

Tableau 2.30 – Diversité thématique avec le graphe Cora

2.7 Bilan

Nous avons mis l'accent, dans ce chapitre, sur le fait que le phénomène TKC représente un sérieux problème pour de nombreux algorithmes de calcul de centralité. Hormis l'algorithme PageRank qui est assez robuste à l'effet TKC, nous avons remarqué que la majorité des algorithmes de calcul de centralité existants sont très vulnérables à ce symptôme. Dans le but de pallier à ce problème, nous avons proposé trois algorithmes fondés sur des principes différents. L'évaluation expérimentale de nos algorithmes a montré les très bonnes performances de l'algorithme DocRank.

Dans la deuxième partie de cette thèse, nous allons nous intéresser à la notion de communauté de manière plus détaillée. Nous abordons notamment la problématique de l'identification de structures de communautés (ISC) dans les graphes de documents.

3

Etat de l'Art sur l'Identification de Structures de Communautés (ISC)

Les réseaux complexes (ou graphes de terrain) possèdent plusieurs caractéristiques qui les distinguent des graphes aléatoires (tels que les graphes d'Erdös-Rényi). L'organisation des nœuds de ces réseaux en groupes appelés communautés est une des propriétés les plus importantes des réseaux complexes. Intuitivement, une communauté est un ensemble de nœuds dans lequel il y a une forte concentration de liens par rapport au nombre de connexions avec les autres nœuds du graphe. Ces dernières années, en particulier, depuis l'article de référence de Girvan et Newman [Girvan and Newman 02], l'identification de telles structures a suscité l'intérêt d'un grand nombre de chercheurs dans diverses disciplines, notamment avec l'entrée en jeu des physiciens et des mathématiciens. Depuis, une panoplie de techniques a été proposée et de nouvelles méthodes continuent d'apparaître régulièrement. Ceci étant, nous pouvons dire que l'ISC est un domaine vaste et interdisciplinaire auquel contribuent plusieurs communautés.

Après une présentation des différentes définitions de la notion de communauté, nous abordons le problème de l'identification de structures de communautés en décrivant les principales approches existantes. Les techniques d'ISC peuvent être classées de différentes façons. Dans le cadre de cette thèse, nous les avons regroupées en deux catégories à savoir les approches génératives (basées sur un modèle génératif) et les approches non génératives. Ces dernières seront d'abord présentées de manière succincte alors que les premières feront ensuite l'objet d'une étude plus détaillée. Enfin, en utilisant un certain nombre de critères que nous avons établis, nous analysons l'adéquation des différentes techniques exposées pour l'identification de communautés dans les graphes de documents.

3.1 Notion de communauté et problème de l'ISC

Dans l'édition 2010 du dictionnaire Larousse, le mot *communauté* est défini par :

> « État, caractère de ce qui est commun à plusieurs personnes : *Une communauté de biens*, *d'intérêts.*
> Ensemble de personnes unies par des liens d'intérêts, des habitudes communes, des opinions ou des caractères communs : *Communauté ethnique*, *linguistiques.*
> Ensemble des citoyens d'un État, des habitants d'une ville ou d'un village. »

Le terme communauté désigne, comme son radical l'indique, un ensemble d'individus ayant une ou plusieurs caractéristiques (propriété, centre d'intérêt, etc.) en *commun*. Cette notion de communauté a fait l'objet de nombreuses études de la part des sociologues depuis plusieurs décennies, notamment dans le cadre de l'analyse des réseaux sociaux [Scott 00]. Les réseaux sociaux sont des structures modélisant les relations sociales (par exemple l'amitié, la collaboration, la parenté, etc.) qui existent entre un ensemble d'individus appelés aussi acteurs. Ces réseaux sont généralement représentés sous forme d'un graphe dans lequel les sommets correspondent aux entités sociales (individus ou acteurs) et les liens correspondent aux relations sociales [Wasserman and Faust 94].

Depuis quelques années, l'utilisation du mot communauté s'est généralisée à d'autres types de graphes et n'est désormais plus réservée aux réseaux sociaux. Nous retrouvons cette extension d'usage dans plusieurs disciplines telles que l'informatique [Dourisboure et al. 07], la physique [Hastings 06] ou encore la biologie [Bornholdt and Schuster 03]. En informatique *par exemple*, la notion de communauté est devenue dès la fin des années 90 très populaire avec le développement du web et de la recherche d'information basée sur les liens hypertextes. Des chercheurs comme Kleinberg [Kleinberg 99a] ou Flake [Flake et al. 00] ont introduit la notion de *communauté web* qui désigne un ensemble de pages web traitant une même thématique ou un même sujet (ou « topic » en anglais).

La définition la plus courante de la notion de communauté dans un graphe (ou réseau) est celle d'un ensemble de sommets ayant beaucoup de liens (i.e. une forte densité des liens) entre eux et peu de liens (i.e. une faible densité des liens) avec les autres nœuds du graphe [Flake et al. 04]. Cette définition étant toutefois très générale, il est souvent nécessaire de l'affiner afin de pouvoir l'utiliser dans la pratique, comme par exemple pour vérifier si un sous-graphe correspond à une communauté ou encore pour mettre en œuvre un outil d'identification automatique de communautés. Malheureusement, il n'existe dans la littérature aucune définition formelle et universelle de ce qu'est une communauté [Fortunato 10]. Nous présentons ci-après plusieurs définitions en les regroupant en trois catégories à savoir celles basées sur la connectivité des nœuds, celles basées sur la proximité des nœuds et celles basées sur une fonction de qualité.

Mais avant de présenter ces définitions, nous tenons à préciser une propriété importante qui doit être vérifiée par toute communauté (ou plus précisément par le sous-graphe correspondant à la communauté). Il s'agit de la propriété de *connexité* qui signifie que le sous-graphe correspondant à la communauté doit être connexe. Cette propriété signifie qu'il existe

un chemin entre tous les nœuds du graphe quelque soit sa longueur et en ignorant l'orientation des arêtes. Dans le cas où le graphe à analyser contient plusieurs composantes connexes (i.e. n'est pas connexe), il faudra alors considérer chaque composante comme étant un graphe à part entière et rechercher des communautés dans chacune de ces composantes. Cependant, pour des raisons de simplification, nous considérons tout au long de cette thèse, que les graphes étudiés sont connexes. De plus, dans les définitions suivantes nous considérons que les graphes sont non-orientés.

3.1.1 Définitions basées sur la connectivité des sommets

Une des premières définitions de la notion de communauté a été proposée par Luce et Perry [Luce and Perry 49] dans le cadre de leurs travaux sur l'analyse des réseaux sociaux. Ils ont ainsi proposé le terme *clique* pour faire référence à un groupe composé d'au moins trois personnes qui se connaissent toutes. Ce terme a ensuite été repris en théorie des graphes pour désigner un sous-graphe complet i.e. un ensemble de sommets deux-à-deux adjacents. La figure 3.1 illustre des exemples de cliques. Sur cette figure, *C1* et *C3* sont des cliques ; C2 n'est par contre pas une clique car ses nœuds ne sont pas tous connectés.

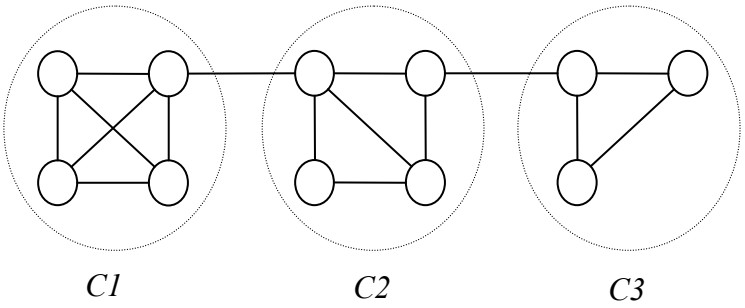


Figure 3.1 – Exemple d'un graphe contenant deux cliques (*C1* et *C3*)

Bien que simple et théoriquement très intéressante, la définition d'une communauté en termes de cliques possède de nombreux inconvénients. Une de ses limites est que dans les graphes réels, il est très rare de rencontrer des cliques de grande taille, on trouve plutôt des triangles qui sont des cliques composés de trois sommets [Fortunato 10]. Un autre problème avec cette définition est qu'elle est très exigeante dans la mesure où elle nécessite la présence de liens entre tous les sommets du sous-graphe. On pourrait raisonnablement considérer que le sous-graphe *C2* sur la figure 3.1 correspond à une communauté même s'il lui manque un lien pour former une clique. Cette idée a été exploitée par de nombreux chercheurs qui ont proposé des notions plus flexibles que celle de clique. Nous présentons ci-dessous quelques unes de ces définitions :

- *n-clique :* c'est un sous-graphe maximal tel que la distance géodésique entre chaque couple de ses sommets est inférieure ou égale à *n* [Alba 73].
- *n-clan :* c'est une n-clique dont le diamètre est inférieur ou égal à *n* [Mokken 79].

- *n-club* : c'est un sous-graphe maximal dont le diamètre est inférieur ou égal à n [Mokken 79].
- *k-plex* : c'est un sous-graphe maximal tel que ses sommets peuvent ne pas être adjacents à k sommets au maximum [Seidman and Foster 78].
- *k-core* : c'est un sous-graphe maximal tel que ses sommets sont adjacents à au moins *k* sommets [Seidman and Foster 78].

Afin d'illustrer ces différentes notions, considérons le graphe de la figure 3.2. Ce graphe contient entre autres les structures suivantes :

- Trois cliques : {1,2,3,4}, {2,4,5}, {5,6,8}.
- Quatre 2-cliques : {1,2,3,4,5}, {2,4,5,6,8}, {5,6,7,8,9}, {6,7,8,9,10}.
- Trois 2-clans : {1,2,3,4,5}, {2,4,5,6,8}, {6,7,8,9,10}.
- Cinq 2-clubs : {1,2,3,4,5}, {2,4,5,6,8}, {5,6,7,8}, {5,6,8,9}, {6,7,8,9,10}.
- Un 3-core : {1,2,3,4}
- Trois 2-plex : {1,2,3,4,5}, {2,4,5,6,8}, {6,7,8,9,10}.

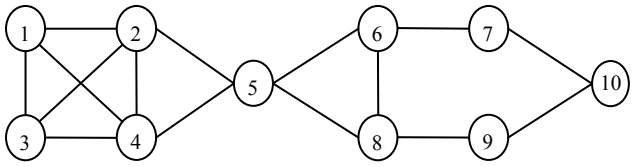


Figure 3.2 – Exemple d'un graphe contenant diverses structures

Une caractéristique commune à ces différentes définitions est le fait qu'elles ne s'intéressent qu'à un seul aspect de la notion de communauté à savoir la forte densité des liens intra-communauté. Or, il est important aussi de considérer l'autre aspect relatif à la faible densité des liens inter-communauté. La prise en compte simultanée de ces deux propriétés a donné lieu à des définitions alternatives de cette notion de communauté. Par exemple, Flake et al. [Flake et al. 00] définissent une communauté comme un ensemble de sommets tel que chaque sommet possède plus de liens vers l'intérieur de la communauté que vers l'extérieur. Raddichi et al. [Radicchi et al. 04] notent cependant que cette définition est très stricte et proposent une définition moins stricte qu'ils appellent *communauté faible* par opposition à la première définition appelée *communauté forte*. Une communauté faible est définie comme un ensemble de sommets dont le nombre de liens internes est supérieur au nombre de liens externes. Formellement, un sous-graphe $C \subset G$ est :

- une communauté faible si : $\sum_{i \in C} d_i^{in}(C) > \sum_{i \in C} d_i^{out}(C)$

- une communauté forte si : $d_i^{in}(C) > d_i^{out}(C), \quad \forall i \in C$

où $d_i^{in}(C)$ est le nombre de liens entre le sommet $i \in C$ et les autres sommets de C ; $d_i^{out}(C)$ est le nombre de liens entre le sommet $i \in C$ et les sommets de G qui ne sont pas dans C.

Il est évident, d'après ces deux définitions, qu'une communauté forte est aussi une communauté faible alors que l'inverse n'est pas vrai. Afin d'illustrer ces deux concepts, considérons le graphe de la figure 3.3. Dans ce graphe, *C3* est une communauté forte, *C1* est une communauté faible alors que *C2* ne correspond à aucune des deux définitions. Notons toutefois qu'il existe d'autres communautés (faibles ou fortes) dans ce graphe que nous n'avons pas indiquées.

3.1.2 Définitions basées sur la similarité des sommets

Une autre définition de la notion de communauté est celle d'un groupe de sommets (ou de points) qui sont à la fois fortement *similaires* entre eux et faiblement *similaires* aux sommets appartenant aux autres groupes [Fortunato 10]. Cette définition, équivalente à celle d'un *cluster* en statistiques et en fouille de données, repose sur l'utilisation d'une mesure de proximité adaptée au type des objets à regrouper. En pratique, cette mesure de proximité peut correspondre soit à une mesure de similarité, soit à une mesure de distance. La littérature sur le clustering contient un grand nombre de travaux sur les mesures de similarité et de distance ; nous présentons ici celles qui sont les plus utilisées.

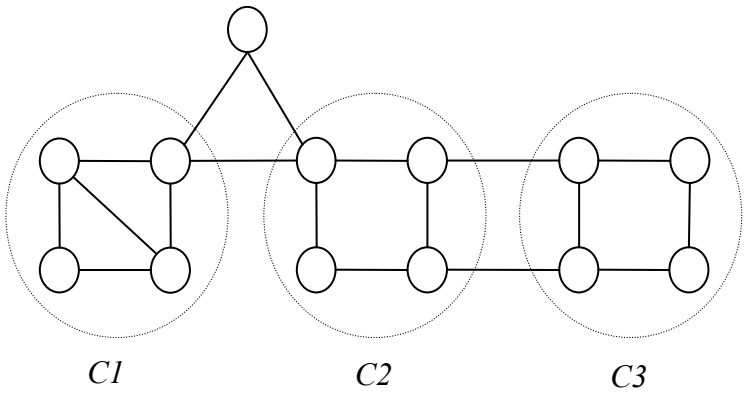


Figure 3.3 – Exemple de graphe contenant une communauté faible (C1) et une communauté forte (C3)

Dans le cas où les objets à comparer peuvent être représentés sous forme de vecteurs à *n* dimensions sur un espace euclidien, on peut alors utiliser des mesures de proximité entre vecteurs telles que la distance euclidienne ou la similarité du cosinus définies respectivement par [Xu and Wunsch 08] :

$$d_{euc}(\mathbf{x},\mathbf{y}) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{3.1}$$

$$s_{\cos}(\mathbf{x},\mathbf{y}) = \frac{\mathbf{y}^t \cdot \mathbf{x}}{\|\mathbf{x}\|\|\mathbf{y}\|} \tag{3.2}$$

où $\mathbf{x},\mathbf{y} \in \mathbb{R}^n$ et $\|\mathbf{x}\| = \sqrt{\sum_{i=1}^{n} x_i^2}$, $\|\mathbf{y}\| = \sqrt{\sum_{i=1}^{n} y_i^2}$ sont les normes L2 des vecteurs $\mathbf{x}$ et $\mathbf{y}$ respectivement.

Si l'on considère par contre que chaque objet est représenté par l'ensemble de ses attributs (par exemple un sommet sera représenté par l'ensemble des sommets qui lui sont adjacents), le calcul de la proximité entre deux objets peut dans ce cas être fait en utilisant des mesures telles que l'indice (ou similarité) de Jaccard défini par [Xu and Wunsch 08] :

$$s_{jac}(x,y) = \frac{|A_x \cap A_y|}{|A_x \cup A_y|} \tag{3.3}$$

où A_x, A_y correspondent respectivement aux ensembles d'attributs des objets x et y.

Il est également possible d'utiliser des mesures de proximité issues de la théorie des graphes telle que la distance du plus court chemin, appelée aussi distance géodésique. Cette métrique est définie comme étant le nombre minimal de liens qu'il faut traverser pour aller d'un sommet à un autre [West 00]. Une mesure de similarité alternative, très utilisée pour le partitionnement de graphes, est celle du nombre de chemins indépendants entre deux nœuds. Elle correspond au nombre minimal de sommets qu'il faut supprimer afin de séparer les deux nœuds [West 00].

En se basant sur le principe de la *marche aléatoire* (cf. section 1.3.1), plusieurs chercheurs ont proposé des mesures de distance entre deux sommets dans un graphe. Par exemple dans [Zhou et al. 07], les auteurs proposent d'utiliser comme mesure de distance le nombre moyen de pas aléatoires nécessaires pour faire un aller-retour entre deux sommets. Les auteurs appellent cette mesure le "average commute time", c'est-à-dire le temps moyen d'un aller-retour entre deux sommets.

3.1.3 Définitions basées sur une fonction de qualité

Une définition plus récente de la notion de communauté est basée sur l'utilisation d'une fonction de qualité. Cette famille de définitions a d'ailleurs reçu l'attention d'un grand nombre de chercheurs depuis la publication en 2004 d'un article par Newman et Girvan [Newman and Girvan 04]. Une fonction de qualité est une fonction qui associe à un sous-graphe S une valeur quantifiant le fait que S correspond à une communauté.

Suite à l'article de référence de Newman, plusieurs fonctions de qualité ont été proposées par les chercheurs. Nous allons cependant nous contenter ici de présenter celle introduite par Newman connue sous le nom de la modularité. Celle-ci repose sur l'idée qu'un sous-graphe forme une communauté si la distribution des liens entre ses nœuds n'est pas due au hasard. Plus précisément, la modularité mesure la divergence de la distribution des liens dans le sous-graphe par rapport à la distribution des liens dans un graphe aléatoire qui ne contient pas de

communautés. Formellement, étant donné un graphe non-orienté $G=(V,E)$ d'ordre *N* et de taille *M*, la modularité d'un sous-graphe $S=(V_S,E_S)$ de *G* est définie par [Newman and Girvan 04] :

$$Q(S)=\frac{|E_S|}{M}-\left(\frac{d_s}{2M}\right)^2 \tag{3.4}$$

où d_S est égale à la somme des degrés des nœuds appartenant à S. Le premier terme de la formule 1.1 correspond à la proportion de liens entre les nœuds du sous-graphe *S*, tandis que le deuxième terme correspond à la proportion de liens dans un graphe aléatoire ayant la même taille et la même distribution des degrés que *S*.

Les fonctions de qualité et en particulier la modularité sont généralement utilisées pour estimer la qualité d'une partition des sommets d'un graphe en communautés. Dans cette optique, la modularité associée à une partition $P=(C_1,\ldots,C_k)$ est égale à la somme de la modularité des communautés qui la composent i.e. :

$$Q(P)=\sum_{i=1}^{k}Q(C_i)=\sum_{i=1}^{k}\left(\frac{|E_{C_i}|}{M}-\left(\frac{d_{C_i}}{2M}\right)\right)^2$$

Cette formule peut également être réécrite de la manière suivante :

$$Q(P)=\sum_{i=1}^{N}\sum_{j=1}^{N}\left[\frac{a_{ij}}{2M}-\frac{d_i d_j}{4M^2}\right]\delta(c_i,c_j) \tag{3.5}$$

où a_{ij} correspond à l'entrée (i,j) de la matrice d'adjacence $\mathbf{A}$, d_i est le degré du nœud *i*, $c_i\in\{1,\ldots,k\}$ indique la communauté du nœud *i*, et δ est la fonction delta de Kronecker.

Cette définition de la modularité concerne les graphes non-orientés. Transposée aux graphes orientés, elle devient [Leicht and Newman 08] :

$$Q(P)=\sum_{i=1}^{n}\sum_{j=1}^{n}\left[\frac{a_{ij}}{m}-\frac{d_i^{in}d_j^{out}}{m^2}\right]\delta(c_i,c_j) \tag{3.6}$$

où d_i^{in} est le degré entrant du nœud *i*, et d_i^{out} le degré sortant du nœud i.

Les valeurs possibles pour la modularité sont dans l'intervalle]-1, 1[. En cas d'absence de structure de communautés, la valeur de la modularité est négative ou nulle, alors qu'une valeur supérieure à 0.3 indique la présence d'une structure de communautés [Newman and Girvan 04].

Considérons le graphe jouet de la figure 3.4 et supposons que l'on souhaite comparer la qualité des partitions $P1=\{\{A,B,C,D\},\{E,F,G,H\}\}$ et $P2=\{\{A,B\},\{C,D,E,F,G,H\}\}$. Le calcul des modularités de *P1* et *P2* indique que $Q(P1)=0.42$ et $Q(P2)=0$. Ce résultat

signifie que la partition P1 est meilleure que P2 et que cette dernière ne correspond pas à une structure de communautés.

La modularité a été initialement proposée par Newman comme solution au problème suivant : étant données *K* partitions des sommets d'un graphe où chaque partition contient un nombre différent de communautés, quelle est la meilleure de ces *K* partitions ? La modularité semblait alors être une réponse très prometteuse à ce problème (qui était jusque-là sans solution satisfaisante) jusqu'à ce que Fortunato et Barthélemy [Fortunato and Barthélemy 07] remettent en cause ce principe de sélection de la meilleure partition en se basant sur la modularité. Ils montrent en effet que la modularité favorise les partitions en communautés de grande taille. Autrement dit, cela signifie que si le graphe contient plusieurs communautés et que certaines de ces communautés sont de petite taille, la modularité tend à favoriser les partitions où les petites communautés ont été regroupées. Fortunato et Barthélemy montrent plus précisément que pour un graphe contenant L liens, la modularité favorise les partitions composées de communautés ayant au moins $\sqrt{L/2}$ liens, même si le graphe contient en réalité des communautés ayant un nombre de liens inférieur à $\sqrt{L/2}$.

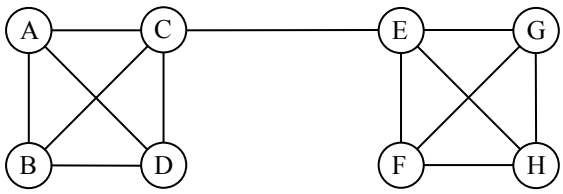


Figure 3.4 - Exemple d'un graphe jouet

3.1.4 L'identification de structure de communautés

L'identification de structure de communautés (ISC), appelée aussi détection ou extraction de communautés, a pour but d'identifier toutes les communautés présentes dans un graphe donné. Une structure de communautés dans un graphe $G=(V,E)$ est un ensemble $S=\{C_1,C_2,\ldots,C_k\}$ tel que : $C_1 \cup C_2 \cup \ldots \cup C_k = V$ et chaque C_i vérifie la définition de communauté considérée. Dans le cas où les ensembles C_i sont disjoints, la structure de communauté est aussi appelée *partition*. Si par contre ils ne sont pas disjoints, on dit qu'il y a *recouvrement* ou *chevauchement* des communautés de la structure. Dans ce cas, la structure de communautés peut parfois être munie d'une matrice dite matrice d'appartenance **M** dans laquelle chaque entrée m_{ij} indique le degré d'appartenance d'un sommet *i* à une communauté *j*.

Mais le fait de devoir assigner chaque nœud à au moins une communauté impose des contraintes sur l'approche à utiliser. En effet, l'utilisation de certaines définitions de la notion de communautés peut conduire à des situations où un nœud n'est assigné à aucune communauté. Prenons un exemple simple afin d'éclaircir ce point. Considérons le graphe de la figure 3.5 et supposons que l'on cherche à identifier sa structure de communautés. Supposons également que les communautés doivent respecter la définition d'une communauté

forte. On remarque que le graphe contient deux communautés fortes *C1={1,2,3,4}* et *C2={6,7,8,9}* mais l'ensemble *S1={C1,C2}* ne constitue pas une structure de communautés puisque le nœud *5* n'appartient à aucune communauté. Si l'on suppose maintenant que les communautés à identifier doivent correspondre à des communautés faibles, on obtiendrait par exemple les deux communautés *C3={1,2,3,4,5} et C4={5,6,7,8,9}*. Dans ce cas, l'ensemble *S2={C3,C4}* constitue bien une structure de communautés. Dans le cadre de cette thèse, nous nous intéressons uniquement aux approches d'ISC qui déterminent une structure de communautés telle que nous venons de la définir.

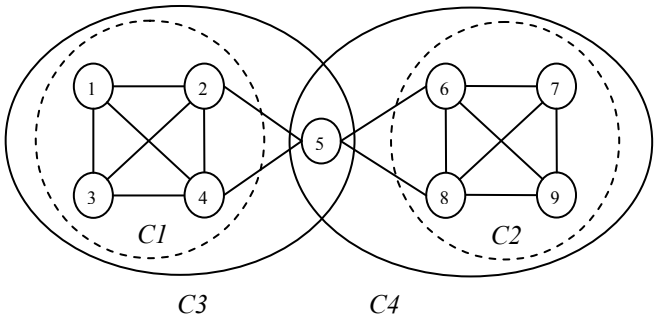


Figure 3.5 – Exemple de graphe contenant deux communautés fortes (*C1* et *C2*) et deux communautés faibles (*C3* et *C4*)

3.2 Approches non génératives pour l'ISC

Nous mettons dans cette catégorie toutes les approches qui ne sont pas basées sur un modèle génératif. A leur tour, ces algorithmes peuvent être classés en plusieurs catégories. Diverses classifications ont été proposées dans la littérature. Elles sont issues de deux domaines différents. La classification non supervisée (ou clustering) de données et le partitionnement de graphes.

3.2.1 Approches basées sur le clustering par partitionnement

Le but du clustering par partitionnement est de regrouper un ensemble d'objets en classes (appelées aussi clusters ou groupes) tel que chaque objet appartient à au moins une classe. Les divers algorithmes appartenant à cette famille d'approches diffèrent par la fonction de qualité utilisée pour évaluer la pertinence d'un regroupement (ou partition) ainsi que par la stratégie utilisée pour rechercher une "bonne" partition en un temps "raisonnable". L'énumération et l'évaluation de toutes les partitions possibles étant un problème NP-complet [Fortunato 10], il est alors nécessaire de recourir à des techniques d'optimisation pour lesquelles une fonction objectif est optimisée jusqu'à l'obtention d'une partition "satisfaisante".

L'algorithme des K-moyennes [Macqueen 67] (voir l'algorithme 3.1) est une technique très populaire de clustering par partitionnement. Il utilise une mesure de distance (resp. de similarité) pour regrouper les objets en minimisant la distance (resp. en maximisant la similarité) intra-cluster. Plus précisément, la fonction objectif optimisée par l'algorithme des K-moyennes est donnée par :

$$J^{K-moy} = \sum_{k=1}^{K} \sum_{i:i \in G_k} d\left(\mathbf{x}_i, \mathbf{c}_k\right) \tag{3.7}$$

où K est le nombre de groupes, G_k représente l'ensemble des objets appartenant au groupe k, $\mathbf{c}_k$ est le centroïde du groupe k (cf. étape 3 de l'algorithme 3.1), $\mathbf{x}_i$ est le vecteur représentant l'objet i, et d est une mesure de distance (tel que la distance euclidienne par exemple). L'algorithme des K-moyennes minimise donc la distance des objets par rapport aux centroïdes des groupes auxquels ils appartiennent.

Comme indiqué par l'algorithme 3.1, l'algorithme des K-moyennes est un algorithme itératif qui procède en deux étapes : assignation des objets aux groupes les plus proches (représentés par leurs centroïdes) puis calcul des centroïdes des nouveaux groupes. Ces deux opérations sont répétées jusqu'à ce qu'un critère de convergence soit satisfait. L'arrêt peut s'effectuer par exemple lorsqu'un nombre maximal d'itération a été atteint ou bien lorsque plus aucun changement ne s'opère sur le contenu des groupes.

Algorithme 3.1 : Algorithme des K-moyennes

Entrée : - un graphe $G = (V, E)$ d'ordre N représenté par sa matrice d'adjacence **A**
- le nombre de communautés K
- une mesure de distance d

Sortie : une partition en communautés $P = \{P_1, \ldots, P_K\}$

début

1. Initialiser les centroïdes (ou centres) : tirer aléatoirement K colonnes de la matrice **A** qui vont jouer le rôle des centroïdes initiaux $\mathbf{c}_1, \ldots, \mathbf{c}_K$
2. Constituer une partition initiale $P^{(0)} = \{G_1^{(0)}, \ldots, G_k^{(0)}\}$ en assignant chaque nœud au centroïde le plus proche
$$G_j^{(0)} \leftarrow \left\{ v_i \in V \mid d\left(\mathbf{a}_{.i}, \mathbf{c}_j\right) = \min_{l=1,\ldots,l} d\left(\mathbf{a}_{.i}, \mathbf{c}_l\right) \right\}$$
3. Calculer les centroïdes $\mathbf{c}_1, \ldots, \mathbf{c}_K$ des nouveaux groupes
$$\mathbf{c}_j \leftarrow \frac{1}{\left|G_j^{(0)}\right|} \sum_{v_m \in G_j^{(0)}} \mathbf{a}_{.m}$$
4. Initialiser le nombre d'itérations $t \leftarrow 1$
5. Constituer une nouvelle partition $P^{(t)} = \{G_1^{(t)}, \ldots, G_k^{(t)}\}$
$$G_j^{(t)} \leftarrow \left\{ v_i \in V \mid d\left(\mathbf{a}_{.i}, \mathbf{c}_j\right) = \min_{l=1,\ldots,l} d\left(\mathbf{a}_{.i}, \mathbf{c}_l\right) \right\}$$
6. Calculer les centroïdes $\mathbf{c}_1, \ldots, \mathbf{c}_K$ des nouveaux groupes
$$\mathbf{c}_j \leftarrow \frac{1}{\left|G_j^{(t)}\right|} \sum_{v_m \in G_j^{(t)}} \mathbf{a}_{.m}$$
7. $t \leftarrow t + 1$
8. **Si** l'algorithme n'a pas convergé **alors** aller à l'étape 5
9. Retourner la partition $P^{(t)}$

fin

3.2.2 Approches basées sur le clustering hiérarchique ascendant

Au lieu de constituer une partition "plate" comme le fait le clustering par partitionnement, le clustering hiérarchique construit plutôt une hiérarchie de partitions représentée sous la forme d'un dendrogramme (voir figure 3.6). Les algorithmes de classification hiérarchiques sont de deux types : les méthodes ascendantes et les méthodes descendantes.

Les méthodes de classification hiérarchique ascendantes supposent au départ que chaque objet (ou nœud) forme un cluster (ou une communauté). Les clusters les plus proches sont ensuite fusionnés de manière récursive jusqu'à l'obtention d'un seul cluster contenant tous les objets. Le calcul de proximité entre deux clusters nécessite l'utilisation d'une mesure de proximité entre les objets (distance euclidienne, coefficient de Jaccard, etc.) ainsi qu'un indice d'agrégation. Il existe dans la littérature plusieurs indices d'agrégation, nous citerons par exemple :

- le lien minimum (single link) : la distance entre deux clusters X et Y est égale à la plus petite distance entre deux objets de ces deux clusters i.e.

$$d_{\min}(X,Y) = \min_{x\in X, y\in Y} d(x,y) \tag{3.8}$$

- le lien moyen (average link) : la distance entre deux clusters X et Y est égale à la moyenne des distances entre les objets de ces deux clusters i.e.

$$d_{avg}(X,Y) = \frac{1}{|X||Y|} \sum_{x\in X, y\in Y} d(x,y) \tag{3.9}$$

- le lien maximum (complete link) : la distance entre deux clusters X et Y est égale à la plus grande distance entre deux objets de ces deux clusters i.e.

$$d_{\max}(X,Y) = \max_{x\in X, y\in Y} d(x,y) \tag{3.10}$$

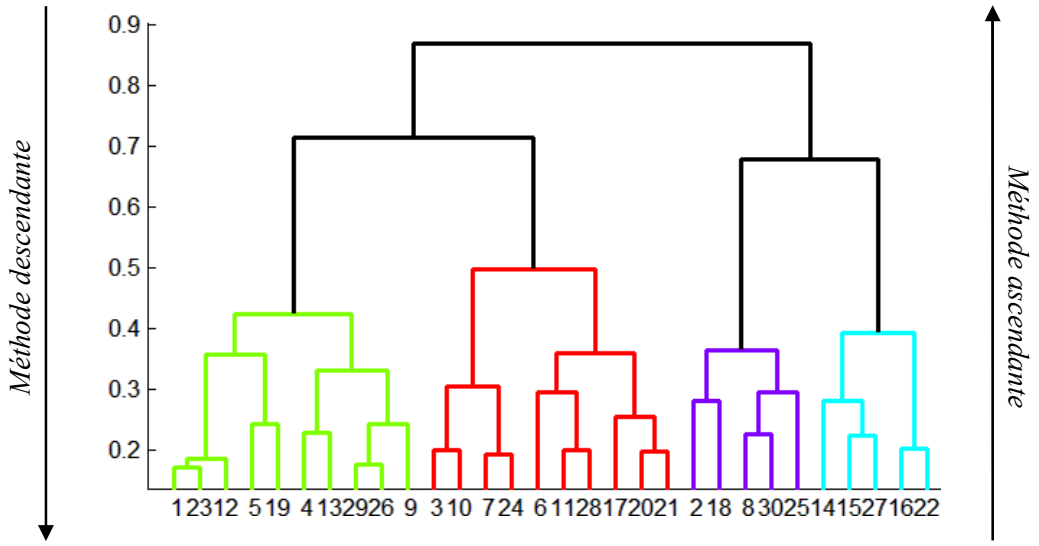


Figure 3.6 - Exemple d'un dendrogramme

Algorithme 3.2 : Algorithme de Donetti et Munoz

Entrée : - un graphe non-orienté $G = (V, E)$ d'ordre N représenté par sa matrice d'adjacence **A**
- le nombre de vecteurs propres (dimension de l'espace de projection) K
- une mesure de proximité *pro* (distance euclidienne ou similarité du cosinus)
- une mesure de similarité entre groupes *sim* (lien minimum ou lien maximum)

Sortie : une partition en communautés P

début

1. Calculer la matrice de Laplace **L**
$$\mathbf{L} \leftarrow diag(\mathbf{A1}) - \mathbf{A}$$
2. Calculer la matrice **U** dont les lignes correspondent aux vecteurs propres $\mathbf{u}_{1.}, \ldots, \mathbf{u}_{K.}$ associés aux K plus petites valeurs propres de la matrice **L**
$$\mathbf{U} \leftarrow eig(\mathbf{L}, K)$$
3. Calculer la matrice des proximités **S** entre toutes les paires de nœuds (i, j) tel que :
$$s_{ij} \leftarrow pro(\mathbf{u}_{.i}, \mathbf{u}_{.j})$$
4. Initialisation : $C \leftarrow \{\{v_1\}, \ldots, \{v_N\}\}$ et $T \leftarrow \{C\}$
5. Identifier dans C les deux communautés C_i et C_j les plus similaires (ou plus proches)
$$(C_i, C_j) \leftarrow \underset{(C_l, C_m) \in C^2}{\arg\max} \{sim(C_l, C_m, \mathbf{S})\}$$
6. Remplacer C_i et C_j dans C par $(C_i \cup C_j)$ et rajouter C à T
7. **Si** $|C| \neq 1$ **alors** aller à l'étape 5
8. Retourner la partition P ayant la plus grande modularité (cf. section 3.1.3)
$$P = \underset{T_i \in T}{\arg\max} \{\mathrm{mod}(G, T_i)\}$$

fin

L'algorithme 3.2 est un algorithme de classification hiérarchique ascendante proposé par Donetti et Munoz [Donetti and Munoz 04] pour l'identification de structures de communautés. Dans leur article, Donetti et Munoz ont testé leur algorithme en utilisant deux mesures de proximité (distance euclidienne et similarité du cosinus) ainsi que deux indices d'agrégation (lien minimum et lien maximum). Ils rapportent que la similarité du cosinus permet une meilleure détection de la structure de communautés que la distance euclidienne. Concernant l'indice d'agrégation, leurs résultats n'ont pas permis de conclure en faveur de l'un des deux indices comparés ; dans certains cas le lien minimal était meilleur alors que dans d'autres cas c'est le lien maximal qui l'était. Pour le choix de la meilleure partition, Donetti et Munoz utilisent la modularité de Newman.

L'originalité de l'approche de Donetti et Munoz est due à la méthode utilisée pour le calcul de proximité entre les N objets. Ce calcul n'est pas fait dans l'espace initial de dimension N (où chaque dimension correspond à un objet) mais plutôt dans un espace de dimension D (où $D < N$) formé par les D premiers vecteurs propres non triviaux (i.e. ceux associés aux D plus petites valeurs propres différentes de zéro) d'une matrice particulière appelée *matrice de Laplace*. Il s'agit d'une matrice dont les propriétés ont été beaucoup

étudiées par les chercheurs dans le cadre du partitionnement de graphes [Aggarwal and Wang 10]. La matrice de Laplace **L** d'un graphe non-orienté *G*, représenté par sa matrice d'adjacence **A**, est définie par :

$$l_{ij} = \begin{cases} d(v_i) & \text{si } i = j \\ -1 & \text{si } i \neq j \text{ et } a_{ij} = 1 \\ 0 & \text{sinon} \end{cases} \quad (3.11)$$

où $d(v_i)$ est le degré du nœud v_i. Donetti et Munoz montrent que la projection des objets sur le nouvel espace permet de faire un "pré-regroupement" des nœuds similaires, ce qui permet d'obtenir de meilleurs indices de proximité.

3.2.3 Approches basées sur le clustering hiérarchique descendant

La classification hiérarchique descendante procède de manière inverse à la classification hiérarchique ascendante. Elle commence au début par une partition constituée d'un seul cluster contenant les *N* objets à regrouper. Ce cluster est ensuite divisé en deux clusters, puis de manière récursive tout cluster est éclaté en deux jusqu'à l'obtention de *N* clusters singletons. Le choix du cluster à diviser à chaque étape représente l'opération la plus importante et la plus délicate des méthodes descendantes. Girvan et Newman [Girvan and Newman 02] (voir l'algorithme 3.3) proposent par exemple de retirer à chaque étape une arête du graphe, ce qui peut parfois avoir pour conséquence de diviser une composante connexe (considérée comme un cluster) en deux composantes connexes. L'arête à retirer correspond à celle qui possède la plus forte centralité d'intermédiarité. Cette dernière est basée sur le même principe que celui de la centralité d'intermédiarité des nœuds que nous avons présentée dans le chapitre 1. La centralité d'intermédiarité d'une arête est égale au nombre total de chemins géodésiques qui utilisent cette arête divisé par le nombre total de chemins géodésiques dans le graphe (cf. étape 2 de l'algorithme 3.3). Enfin pour le choix de la meilleure partition, Newman et Girvan utilisent la modularité.

Pour plus de détails sur les méthodes de clustering, le lecteur est invité à consulter des ouvrages de référence tel que [Jain and Dubes 98][Xu and Wunsch 08][Jain 10].

Algorithme 3.3 : Algorithme de Newman et Girvan

Entrée : - un graphe non-orienté $G=(V,E)$ d'ordre N

Sortie : une partition en communautés P

début

1. Initialisation : $T=\{\{v_1,\ldots,v_N\}\}$, $G'=G$
2. Calculer la centralité d'intermédiarité pour chaque arête e_i du graphe G'

$$C^{\text{int}}(e_i)=\sum_{j=1}^{n}\sum_{k=1}^{n}\frac{g_{jk}(e_i)}{g_{jk}}$$

3. Retirer du graphe G' l'arête e_m ayant la plus grande centralité d'intermédiarité

$$E'=E' \setminus \left\{e_m \in E' \mid e_m=\arg\max_{e_j\in E'} C^{\text{int}}(e_j)\right\}$$

4. Identifier l'ensemble $C=\{C_1,\ldots,C_l\}$ de toutes les composantes connexes du graphe G'
5. **Si** $C \notin T$ **alors** rajouter C à T
6. **Si** $|E'|\neq 0$ **alors** aller à l'étape 2
7. Retourner la partition P ayant la plus grande modularité

$$P=\arg\max_{T_i\in T}\{\text{mod}(G,T_i)\}$$

fin

3.2.4 Approches basées sur le partitionnement de graphe

Le partitionnement de graphe est un problème auquel s'est intéressée l'informatique depuis plusieurs décennies notamment dans le cadre de la mise en œuvre d'architectures parallèles [Schaeffer 07]. De manière générale, le problème du partitionnement de graphe consiste à diviser un graphe G en plusieurs sous-graphes $S_1,\ldots,S_k$ (correspondants à des communautés) tel que le nombre total de liens entre les différents sous-graphes (nombre appelé aussi *taille de la coupe*) soit minimal. En pratique, la plupart des approches de partitionnement de graphes procèdent par une division du graphe en deux sous-graphes (ou communautés) puis par un partitionnement récursif des deux sous-graphes ainsi obtenus. L'arrêt du partitionnement se fait lorsque le nombre de communautés souhaité est atteint.

L'algorithme 3.4 décrit les différentes étapes de l'algorithme de bissection spectrale [Barnes 82], un algorithme très populaire pour le partitionnement de graphes. Comme son nom l'indique, l'algorithme est basé sur le calcul de vecteurs propres, et plus précisément, du premier vecteur propre non trivial de la matrice de Laplace. Ce vecteur est aussi connu sous le nom de vecteur de Fiedler. L'algorithme de bissection spectrale est basé sur l'idée que le vecteur de Fiedler représente une bonne solution au problème d'identification de la partition d'un graphe G en deux sous-graphes (de même ordre ou presque) S_1 et S_2 tel que la taille de la coupe R soit minimale. La taille de la coupe R peut en effet être exprimée par [West 00] :

$$R=\frac{\mathbf{x}^T\mathbf{L}\mathbf{x}}{4} \tag{3.12}$$

où **L** est la matrice de Laplace et **x** est un vecteur indicateur tel que $x_i = +1$ si $v_i \in S_1$ et $x_i = -1$ si $v_i \in S_2$. Nous renvoyons le lecteur intéressé par les techniques de partitionnement de graphes vers des références comme [Cook and Holder 06][Schaeffer 07].

Algorithme 3.4 : Algorithme de bissection spectrale

Entrée : - un graphe non-orienté $G = (V, E)$ d'ordre *N* représenté par sa matrice d'adjacence **A**

Sortie : une partition en deux communautés *P*

début

1. Calculer la matrice de Laplace **L**

$$\mathbf{L} = diag(\mathbf{A1}) - \mathbf{A}$$

2. Calculer le vecteur propre **u** associé à la plus petite valeur propre non triviale (i.e. différente de zéro) de la matrice **L**

$$\mathbf{u} = eig(\mathbf{L}, 1)$$

3. Retourner la partition $P = \{C_1, C_2\}$ tel que

$$C_1 = \{v_i \mid u_i > 0\} \text{ et } C_2 = \{v_j \mid u_j < 0\}$$

fin

3.2.5 Autres approches

L'utilisation de la modularité comme fonction objectif à optimiser est une idée qui a été explorée par plusieurs chercheurs. Le but étant de trouver parmi toutes les partitions possibles celle qui possède la meilleure modularité. Newman [Newman 04] par exemple a proposé un algorithme glouton d'optimisation de la modularité. Il s'agit d'un algorithme de classification hiérarchique ascendante où à chaque itération, l'algorithme réalise la fusion des communautés (ou clusters) qui permet d'augmenter le plus (ou de diminuer le moins) la modularité.

Pons et Latapy [Pons and Latapy 05][Pons 07] ont proposé un algorithme de classification hiérarchique ascendante dans lequel le calcul des distances entre les nœuds (ou objets) repose sur le principe de la marche aléatoire. Plus précisément, la distance entre deux nœuds *i* et *j* est exprimée par la différence entre le comportement de deux marcheurs aléatoires commençant respectivement aux nœuds *i* et *j* et effectuant une marche de longueur fixe. Pour le calcul de proximité entre deux communautés, Pons et Latapy généralisent la distance proposée pour les nœuds à des ensembles de nœuds (i.e. des communautés). Une fonction de qualité basée sur la similarité des nœuds est utilisée à la fin pour le choix de la meilleure partition (i.e. meilleure structure de communautés).

En se basant sur une nouvelle définition par émergence des communautés, Bennouas [Bennouas 05] et Bouklit [Bouklit 06] ont proposé deux modèles (gravitationnel et intentionnel) pour la détection de communautés dans les graphes du web. Leurs modèles considèrent que les pages web (ou objets à regrouper) représentent des particules et que les liens représentent des forces gravitationnelles. L'identification de communautés se fait de

manière itérative où à chaque itération, les particules se déplacent dans un espace tridimensionnel vers des objectifs en appliquant deux règles : règle d'exclusion et règle de fusion. Cette procédure itérative s'arrête lorsque les communautés obtenues (correspondant à des groupes de particules proches les unes des autres) sont de bonne qualité conformément à une fonction de qualité inspirée de la modularité de Newman.

3.3 Approches génératives pour l'ISC

L'utilisation des modèles génératifs pour l'ISC est un axe de recherche qui a été peu exploré par rapport aux approches classiques telles que les méthodes hiérarchiques ou encore celles basées sur la modularité de Newman. Il existe d'ailleurs de nombreux travaux de synthèse sur les approches non génératives pour l'ISC (par exemple [Schaeffer 07], [Porter et al. 09] ou [Fortunato 10]) contrairement aux approches génératives dont le domaine semble moins bien cerné. C'est pourquoi nous avons essayé de regrouper dans cette partie de la thèse les principaux modèles génératifs existants pour l'ISC. Nous avons également tenu à les présenter de manière détaillée et homogène (pour la notation) afin de faciliter leur compréhension. Pour chacun des modèles, nous décrivons son processus génératif, sa représentation graphique ainsi que l'algorithme d'estimation de ses paramètres.

Avant de présenter les modèles génératifs existants pour l'ISC, nous décrivons d'abord les principales notions nécessaires à la compréhension de ce type de modèles. Le lecteur familier avec la théorie des probabilités et les modèles génératifs pourra cependant passer directement à la section 3.3.2. Nous invitons par ailleurs le lecteur souhaitant approfondir ses connaissances sur les modèles génératifs à consulter l'excellent ouvrage de référence de Christopher Bishop [Bishop 07].

3.3.1 Introduction aux modèles génératifs

Un modèle génératif suppose que les données observées (images, textes, graphes, etc.) sont le produit ou le résultat d'un processus aléatoire [Bishop 07]. Un modèle génératif est composé de deux parties. La première partie est un modèle probabiliste qui se présente sous la forme d'une distribution paramétrique décrivant le processus par lequel ont été générées les données. Cette modélisation consiste à utiliser des variables aléatoires et à définir des liens de dépendance entre ces variables. Les liens entre variables, appelés aussi liens de causalité, traduisent les hypothèses sur lesquelles est basé le modèle. Une représentation graphique du modèle est souvent utilisée pour faciliter sa lecture et sa compréhension ; on parle aussi dans ce cas de modèle graphique. Quant à la deuxième partie, elle correspond à une méthode d'estimation qui calcule les valeurs optimales des paramètres du modèle. Une telle méthode permet d'apprendre de manière automatique les paramètres du modèle à partir des données.

Afin d'illustrer le principe des modèles génératifs, nous allons utiliser deux modèles jouets *M1* et *M2*. Nous supposons que l'ensemble des données observées $D = \{x_1, \ldots, x_N\}$ correspond à un ensemble de *N* nombres entiers compris entre 1 et *M*.

Dans le modèle *M1*, les nombres observés sont supposés être générés par une seule distribution multinomiale. Cette dernière est généralement utilisée pour modéliser des variables à valeurs discrètes. Pour notre exemple, le nombre de valeurs possibles étant égal à *M*, la distribution multinomiale peut être vue comme un dé à *M* faces tel que la probabilité de chaque face est donnée par un paramètre de la distribution. Le modèle *M1* suppose donc que l'ensemble *D* est le résultat de *N* tirages indépendants selon une loi multinomiale i.e. $X \sim \text{Mult}\left(N,\left(\pi_1,\ldots,\pi_M\right)\right)$ où π est un vecteur de paramètres tel que $\sum_{m=1}^{M}\pi_m = 1$.

La figure 3.7a est une représentation graphique du modèle *M1* dans laquelle les variables aléatoires sont représentées par des cercles (appelés aussi des nœuds) et les paramètres du modèle par des points. La figure 3.7b est une représentation plus compacte du modèle *M1* où les *N* variables X_1,.., X_N sont représentées par un rectangle contenant un seul nœud *X* ainsi qu'une inscription du nombre *N* au bas de ce rectangle.

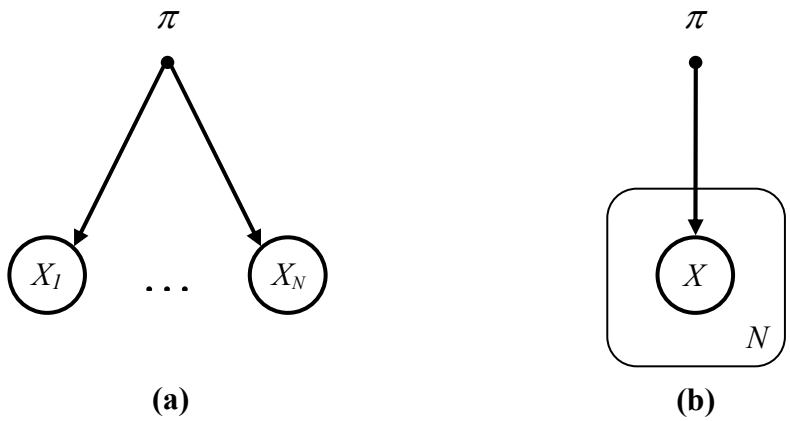


Figure 3.7 - (a) Représentations graphique standard du modèle *M1* ; (b) Représentation graphique compacte du modèle *M1*

Dans ce modèle, la *probabilité a priori* de tirer un nombre *m* est :

$$p\left(X = m\ ;\pi\right) = \pi_m \tag{3.13}$$

Le modèle *M1* est basé sur l'hypothèse que les nombres $\{x_i\}$ sont indépendants les uns des autres. La probabilité de l'ensemble des données observées *D*, appelée aussi *vraisemblance* des données, est alors :

$$\begin{aligned} p\left(D\ ;\pi\right) &= p\left(x_1,\ldots,x_N\ ;\pi\right) \\ &= \text{Mult}\left(X\ ;M,\pi\right) \\ &= \frac{N!}{\prod_{m=1}^{M} N_m!}\prod_{m=1}^{M} p\left(X = m\ ;\pi\right)^{N_m} \\ &= \frac{N!}{\prod_{m=1}^{M} N_m!}\prod_{m=1}^{M} \pi_m^{N_m} \end{aligned} \tag{3.14}$$

où N_m est le nombre de fois que le nombre *m* apparaît dans l'ensemble *X*.

Nous allons maintenant nous intéresser au calcul des paramètres $(\pi_1,\ldots,\pi_K)$ du modèle *M1*. Il s'agit en fait de trouver le modèle qui a la plus forte probabilité d'avoir généré l'ensemble *D*. L'estimation par maximum de vraisemblance (maximum likelihood estimation ou MLE) est une technique classique pour résoudre ce problème. Elle considère la vraisemblance comme une fonction à optimiser par rapport aux paramètres du modèle. Si on note par $L(\pi \mid D) = p(D\,;\pi)$ la fonction de vraisemblance, le but est de trouver le vecteur π tel que $L(\pi \mid D)$ soit maximale i.e.

$$\pi^{MLE} = \arg\max_{\pi} L(\pi \mid D) \tag{3.15}$$

En pratique, il est généralement plus facile de maximiser la *log-vraisemblance* que de maximiser la vraisemblance. Le *log* étant une fonction monotone, maximiser la log-vraisemblance équivaut donc à maximiser la vraisemblance. Un autre intérêt d'utiliser la log-vraisemblance est que le produit de probabilités très petites dans la fonction vraisemblance, peut être inférieur à la limite de représentation sur les ordinateurs. Une somme de *logs* permet par contre d'éviter une telle situation [Bishop 07].

Pour le modèle M_1, la log-vraisemblance *LL* des données est :

$$LL = \sum_{m=1}^{M} N_m \log \pi_m + cst \tag{3.16}$$

où *cst* désigne une constante que l'on négligera par la suite.

Avant de maximiser la log-vraisemblance, il est nécessaire de rajouter des *multiplicateurs de Lagrange* pour que la condition de normalisation $\sum_{m=1}^{M} \pi_m = 1$ soit respectée. On obtient ainsi le Lagrangien suivant :

$$H = \sum_{m=1}^{M} N_m \log \pi_m + \lambda\left(1 - \sum_{m=1}^{M} \pi_m\right) \tag{3.17}$$

où λ est un multiplicateur de Lagrange.

En résolvant l'équation $\frac{\partial H}{\partial \pi_m} = 0$, on obtient la valeur du paramètre π_m qui maximise la log-vraisemblance :

$$\pi_m = \frac{N_m}{N} \tag{3.18}$$

La probabilité de tirer le nombre *m* est donc proportionnelle au nombre de fois que l'entier *m* apparaît dans la liste *D*.

L'estimation des paramètres du modèle M1 est simple car le modèle ne contient que des variables observables. Cependant, comme nous allons le voir dans les sections suivantes, les modèles réels contiennent généralement des variables observables et des variables non observables ; on parle alors de modèles à variables latentes (ou à variables cachées). Pour illustrer ce type de modèles, considérons le modèle *M2* suivant qui suppose que l'ensemble *D*

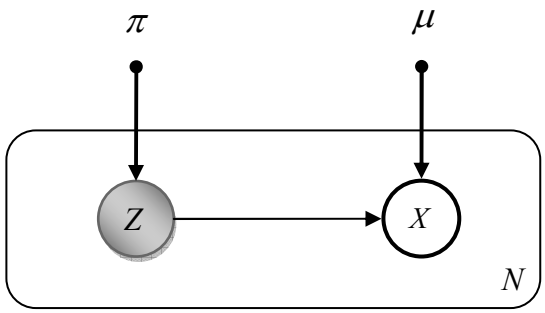


Figure 3.8 - Représentation graphique du modèle *M2*

a été généré par *K* distributions multinomiales différentes, c'est-à-dire que chaque nombre a été généré par un dé parmi *K*. Le processus génératif du modèle *M2* est donc :

Pour $n = 1,\ldots,N$

1. Choisir un dé $Z_n \sim \text{Mult}\left(1,(\pi_1,\ldots,\pi_K)\right)$ où π est un vecteur de paramètres tel que $\sum_{k=1}^{K}\pi_k = 1$.
2. Conditionnellement à Z_n, tirer un nombre $x_n \sim \text{Mult}\left(1,(\mu_{1Z_n},\ldots,\mu_{MZ_n})\right)$ où μ est une matrice de paramètres de dimension $K \times M$ telle que $\forall k \in \{1,\ldots,K\}, \sum_{m=1}^{M}\mu_{mk} = 1$.

Nous noterons par Θ l'ensemble des paramètres de ce modèle i.e. :

$$\Theta = \left\{ \left(\pi_k\right)_{k=1,\ldots,K}, \left(\mu_{mk}\right)_{\substack{m=1,\ldots,M \\ k=1,\ldots,K}} \right\}$$

La représentation graphique de ce modèle est donnée par la figure 3.8. Les variables $\{Z_n\}$ sont représentées par des cercles/nœuds grisés pour indiquer qu'il s'agit de variables non observables (ou cachées) ; le dé ayant servi à générer un nombre donné est une information inconnue. Ce modèle est basé sur les hypothèses suivantes :

i) La distribution jointe d'un nombre x_i et du dé z_i qui a servi a son tirage :

$$\begin{aligned} p\left(X = x_i, Z = z_i \;;\Theta\right) &= p\left(Z = z_i \;;\Theta\right) p\left(X = x_i \mid Z = z_i \;;\Theta\right) \\ &= \pi_{z_i}\mu_{x_i z_i} \end{aligned} \tag{3.19}$$

ii) Les nombres observés $\{x_i\}$ sont tirés de manière indépendante les uns des autres, d'où une vraisemblance de l'ensemble *D* égale à :

$$\begin{aligned} L &= p\left(\left(x_1,\ldots,x_N\right) \;;\Theta\right) \\ &= \prod_{i=1}^{N} p\left(X = x_i \;;\Theta\right) \\ &= \prod_{m=1}^{M} p\left(X = m \;;\Theta\right)^{N_m} \\ &= \prod_{m=1}^{M}\left(\sum_{k=1}^{K} p\left(X = m, Z = k \;;\Theta\right)\right)^{N_m} \\ &= \prod_{m=1}^{M}\left(\sum_{k=1}^{K}\pi_k\mu_{mk}\right)^{N_m} \end{aligned} \tag{3.20}$$

où N_m est le nombre de fois que l'entier m a été observé. Le passage de la ligne 3 à la ligne 4 dans 1.11 a été fait en utilisant la *règle de la somme*. Celle-ci indique que [Bishop 07] :

$$p(A) = \begin{cases} \sum_B p(A,B) & \text{si } B \text{ est une variable discrète} \\ \int p(A,B)\,dB & \text{si } B \text{ est une variable continue} \end{cases}$$

Pour le passage de la ligne 4 à la ligne 5, nous avons utilisé la formule 1.10.

La log-vraisemblance des données est alors égale à :

$$LL = \sum_{m=1}^{M} N_m \log\left(\sum_{k=1}^{K} (\pi_k \mu_{mk}) \right) \tag{3.21}$$

Cette log-vraisemblance possède une forme plus complexe que celle associée au modèle M_1. Il n'est en effet pas possible de l'optimiser car le log est "bloqué" par la somme et ne peut simplifier sa forme. Cela nous donne une expression de la log-vraisemblance qui ne peut généralement pas être optimisée de manière analytique [McLachan and Krishnan 97]. Pour résoudre ce problème, Dempster et al. [Dempster et al. 77] ont proposé l'algorithme EM (Expectation Maximization) permettant d'estimer les paramètres d'un modèle contenant des variables cachées. Cet algorithme est basé sur la décomposition suivante de la log-vraisemblance des données observées [Bishop 07] :

$$\log p(D\,;\Theta) = LB(q(.)\,;\Theta) + KL(q(.) \,\|\, p(.\,|\,D\,;\Theta)) \tag{3.22}$$

où

$$LB(q(.)\,;\Theta) = \sum_Z q(Z) \log\left\{ \frac{p(D,Z\,;\Theta)}{q(Z)} \right\}$$

et

$$KL(q(.) \,\|\, p(.\,|\,D\,;\Theta)) = -\sum_Z q(Z) \log\left\{ \frac{p(Z\,|\,D\,;\Theta)}{q(Z)} \right\}$$

$KL(q \,\|\, p)$ est appelée la divergence de Kullback-Leibler. C'est une distance entre deux distributions de probabilité : sa valeur est comprise entre 0 et 1 et est égale à 0 lorsque $q = p$. LB est une borne inférieure à la log-vraisemblance qui égale cette dernière lorsque $KL(q \,\|\, p) = 0$. Intuitivement, il suffit de choisir la distribution $q(Z)$ tel que :

$$q(Z) = p(Z\,|\,D\,;\Theta) \tag{3.23}$$

pour que la divergence de Kullback-Leibler soit égale à zéro. La distribution $p(Z\,|\,D\,;\Theta)$ est appelée *distribution a postériori* des variables cachées.

Le principe de l'algorithme EM est en fait de trouver les valeurs des paramètres Θ qui maximisent la borne inférieure

$$LB\left(p\left(Z \mid D\ ;\Theta^{old}\right)\ ;\Theta\right) = \sum_{Z} p\left(Z \mid D\ ;\Theta\right) \log p\left(D, Z\ ;\Theta\right) + cst$$
$$= \mathbf{E}_Z\left[\log p\left(D, Z\ ;\Theta\right)\right] + cst$$

qui est plus simple à optimiser que la log-vraisemblance. $\mathbf{E}_Z$ désigne l'espérance par rapport aux variables cachées $\{Z_i\}$ conditionnellement aux données observées. La distribution $p\left(D, Z\ ;\Theta\right)$ est appelée *vraisemblance des données complètes* (i.e. celle des données observées et des valeurs des variables cachées) par opposition à la distribution $p\left(D\ ;\Theta\right)$ qui est appelée *vraisemblance des données incomplètes* (i.e. celle des données observées uniquement). L'algorithme EM consiste à initialiser les paramètres du modèle par des valeurs initiales puis à appliquer de manière itérative les deux étapes suivantes : (i) l'étape E (Expectation), dans laquelle il calcule une borne inférieure à la log-vraisemblance qui est égale à l'espérance de la log-vraisemblance des données complètes (d'où le nom de cette étape) ; (ii) l'étape M (Maximization), dans laquelle la borne inférieure est maximisée pour trouver les valeurs des paramètres qui maximisent la log-vraisemblance [McLachan and Krishnan 97]. L'arrêt de l'algorithme peut se faire par exemple lorsque l'augmentation de log-vraisemblance entre deux itérations consécutives est inférieure à un certain seuil ou bien lorsqu'un nombre d'itérations maximal est atteint. Notons également que le maximum calculé par l'algorithme EM est un maximum local et non pas global. La qualité de la solution trouvée par l'algorithme dépend en effet des valeurs initiales des paramètres du modèle.

En revenant à l'exemple du modèle M_2 et en supposant que l'on connaît le dé qui a servi au tirage de chacun des nombres x_i, la log-vraisemblance des données complètes est :

$$\begin{aligned} LL^C &= \log\left(D, Z\ ;\Theta\right) \\ &= \sum_{n=1}^{N} \log p\left(X = x_n, Z = z_n\ ;\Theta\right) \\ &= \sum_{n=1}^{N} \mathbf{1}_{\{x_n = m, z_n = k\}} \log p\left(X = m, Z = k\ ;\Theta\right) \\ &= \sum_{n=1}^{N} \sum_{m=1}^{M} \sum_{k=1}^{K} \mathbf{1}_{\{x_n = m, z_n = k\}} \log\left(\pi_k \mu_{mk}\right) \end{aligned} \tag{3.24}$$

La notation $\mathbf{1}_F$ correspond à la fonction indicatrice (appelée aussi fonction caractéristique) définie par :

$$\mathbf{1}_F = \begin{cases} 1 & \text{si } F \text{ est vrai ;} \\ 0 & \text{sinon} \end{cases}$$

En supposant que les paramètres du modèle sont connus et fixés à des valeurs Θ^{old}, la distribution a posteriori des variables cachées $p\left(Z \mid D\ ;\Theta^{old}\right)$ est obtenue en appliquant la formule de Bayes :

$$
\begin{aligned}
p\left(Z \mid D\ ;\Theta^{old}\right) &= \frac{p\left(D, Z\ ;\Theta^{old}\right)}{p\left(D\ ;\Theta^{old}\right)} \\
&= \frac{\prod_{i=1}^{N} p\left(X = x_i, Z = z_i\ ;\Theta^{old}\right)}{\prod_{i=1}^{N} p\left(X = x_i\ ;\Theta^{old}\right)} \\
&= \prod_{i=1}^{N} \frac{p\left(X = x_i, Z = z_i\ ;\Theta^{old}\right)}{p\left(X = x_i\ ;\Theta^{old}\right)} \\
&= \prod_{i=1}^{N} p\left(Z = z_i \mid X = x_i\ ;\Theta^{old}\right)
\end{aligned}
\tag{3.25}
$$

Nous remarquons que la distribution jointe a posteriori des variables cachées peut être factorisée en un produit des probabilités a posteriori de chaque variable cachée. Cela indique que les variables Z_i sont indépendantes les unes des autres conditionnellement aux données observées. Pour $k = 1, \ldots, K,\ m = 1, \ldots, M$, cette distribution est alors donnée par :

$$
\begin{aligned}
\omega_{km} &= p\left(Z = k \mid X = m\ ;\Theta^{old}\right) \\
&= \frac{p\left(X = m, Z = k\ ;\Theta^{old}\right)}{p\left(X = m\ ;\Theta^{old}\right)} \\
&= \frac{\pi_k^{old} \mu_{mk}^{old}}{\sum_{t=1}^{K} \pi_t^{old} \mu_{mt}^{old}}
\end{aligned}
\tag{3.26}
$$

L'espérance conditionnelle de la log-vraisemblance des données complètes est donc :

$$
\begin{aligned}
Q &= \mathbf{E}_Z\left[LL^C\right] \\
&= \mathbf{E}_Z\left[\sum_{n=1}^{N}\sum_{m=1}^{M}\sum_{k=1}^{K} \mathbf{1}_{\{x_n = m, z_n = k\}} \log\left(\pi_k \mu_{mk}\right)\right] \\
&= \sum_{n=1}^{N}\sum_{m=1}^{M}\sum_{k=1}^{K} \mathbf{1}_{\{x_n = m\}} \mathbf{E}_{Z_n}\left[\mathbf{1}_{\{z_n = k\}}\right] \log\left(\pi_k \mu_{mk}\right) \\
&= \sum_{n=1}^{N}\sum_{m=1}^{M}\sum_{k=1}^{K} \mathbf{1}_{\{x_n = m\}} p\left(Z_n = k \mid X_n = m\ ;\Theta^{old}\right) \log\left(\pi_k \mu_{mk}\right) \\
&= \sum_{n=1}^{N}\sum_{m=1}^{M}\sum_{k=1}^{K} \mathbf{1}_{\{x_n = m\}} \omega_{km} \log\left(\pi_k \mu_{mk}\right) \\
&= \sum_{m=1}^{M}\sum_{k=1}^{K} N_m \omega_{km} \log\left(\pi_k \mu_{mk}\right)
\end{aligned}
\tag{3.27}
$$

où N_m est le nombre d'occurrences de l'entier *m* (i.e. le nombre de fois qu'il a été observé). Le passage de la ligne 3 à la ligne 4 dans 1.18 a été fait en utilisant la propriété que l'espérance de la fonction indicatrice $\mathbf{1}_F$ est égale à la probabilité de *F* (cf. définition de la fonction indicatrice).

Avant de maximiser l'équation l'espérance conditionnelle *Q*, il est nécessaire de rajouter des multiplicateurs de Lagrange pour prendre en compte les contraintes de normalisation $\sum_{k=1}^{K}\pi_k = 1$ et $\forall m \in \{1,\ldots,M\}, \sum_{k=1}^{K}\mu_{km} = 1$. On obtient ainsi le Lagrangien suivant :

$$H = \sum_{m=1}^{M}\sum_{k=1}^{K} N_m \omega_{km} \log\left(\pi_k \mu_{mk}\right) + \lambda\left(1 - \sum_{k=1}^{K}\pi_k\right) + \sum_{m=1}^{M}\sigma_m\left(1 - \sum_{k=1}^{K}\mu_{km}\right) \tag{3.28}$$

où λ et $(\sigma_1,\ldots,\sigma_K)$ sont les multiplicateurs de Lagrange.

L'équation de ré-estimation de π_k obtenue en résolvant l'équation $\frac{\partial H}{\partial \pi_k} = 0$ est :

$$\pi_k^{new} = \frac{1}{N}\sum_{m=1}^{M} N_m \omega_{km} \tag{3.29}$$

Cela signifie que la probabilité a priori de choisir le dé *k* est proportionnelle au nombre d'entiers qui ont été tirés avec ce dé.

De même, en résolvant l'équation $\frac{\partial H}{\partial \mu_{km}} = 0$, on obtient l'équation de ré-estimation suivante :

$$\mu_{km}^{new} = \frac{N_m \omega_{km}}{\sum_{t=1}^{K} N_m \omega_{tm}} \tag{3.30}$$

Comme nous allons le voir dans la suite, il existe parfois des cas où la distribution a postériori des variables cachées $p(Z \mid D\,;\Theta)$ ne peut être calculée. Cela arrive en général lorsque les variables Z_i ne sont pas indépendantes les unes des autres. Dans ce cas, il devient alors nécessaire de recourir à des techniques qui permettent de calculer une distribution approchée de la distribution a posteriori. Il existe deux familles de méthodes pour arriver à cette fin [Bishop 07]: les méthodes stochastiques et les méthodes déterministes. Pour la première famille, la méthode la plus connue est celle de Monté-Carlo qui consiste à faire de l'échantillonnage. Concernant la deuxième famille, l'inférence variationnelle est sans doute la méthode la plus emblématique. Nous l'utiliserons dans la suite de cette thèse. L'inférence variationnelle part de la décomposition de la log-vraisemblance en deux termes : une borne inférieure et une divergence de Kullback-Leibler. Le principe de l'inférence variationnelle est de chercher une distribution paramétrique *q*(Z) qui soit une bonne approximation de la distribution a posteriori *p*(Z|D). L'inférence variationnelle suppose alors que la distribution q(Z) peut être factorisée de telle sorte que :

$$q(Z) = \prod_{i=1}^{N} q(Z_i) \tag{3.31}$$

puis calcule les paramètres de chaque distribution $q(Z_i)$ en maximisant la borne inférieure *LB* (cf. eq. 1.13). L'inférence variationnelle nous indique que la distribution $q(Z_i)$ qui maximise la borne inférieure est donnée par :

$$\log q(Z_i) = E_{Z\setminus^i}\left[\log p(D,Z)\right] + cst \tag{3.32}$$

où $Z\setminus^i$ correspond à l'ensemble des variables Z moins la variable Z_i.

L'algorithme EM est ensuite utilisé pour l'estimation des paramètres du modèle. Ainsi, lors de l'étape E, il calcule la distribution $q(Z)$ puis à l'étape M, il calcule les valeurs des paramètres qui maximisent la borne inférieure à la log-vraisemblance.

3.3.2 Approche basée sur le modèle de mélange de multinomiales

Newman et Leicht [Newman and Leicht 07] ont récemment proposé un modèle probabiliste pour l'identification de structures de communautés. Il s'agit d'un modèle similaire au modèle de mélange de multinomiales (multinomial mixture model) utilisé par Nigam et al. [Nigam et al. 00] dans le cadre d'une application de classification supervisée de textes. En raison de sa simplicité et de ses bonnes performances, le modèle de mélange de multinomiales a reçu beaucoup d'attention de la part des chercheurs dans le domaine de la fouille de textes. A titre d'exemple, la thèse de Rigouste [Rigouste 06] est consacrée à l'étude de ce modèle dans un contexte de classification non supervisée (ou clustering) de données textuelles.

Le modèle de Mélange de Newman et Leicht (MNL) est un modèle génératif de graphes qui considère qu'un sommet comme est un ensemble de liens (vers d'autres sommets), et que cet ensemble de liens est déterminé par la communauté à laquelle appartient ce sommet. Le modèle MNL suppose ainsi que les N sommets d'un graphe G représenté par sa matrice d'adjacence **A** sont générés par le processus suivant :

- Pour chaque sommet i, $i = 1 \ldots N$ faire :

1. Choisir une communauté $c_i \sim \text{Mult}(1,(\pi_1,\ldots,\pi_K))$ où π est un vecteur de paramètres tel que $\sum_{j=1}^{K} \pi_k = 1$.
2. Conditionnellement à c_i, tirer un vecteur *de* N_i sommets $s_i \sim \text{Mult}\left(N_n,(\mu_{1c_n},\ldots,\mu_{Nc_n})\right)$ où μ est une matrice de paramètres de dimension $N \times K$ telle que $\forall k \in \{1,\ldots,K\}, \sum_{j=1}^{N} \mu_{jk} = 1$.
3. Pour $j = 1 \ldots N_n$, générer un lien entre le nœud courant (i.e. le nœud i) et le nœud s_{nj}.

Nous noterons par Θ l'ensemble des paramètres du modèle MNL i.e. :

$$\Theta = \left\{ \left(\pi_k\right)_{k=1,\ldots,K}, \left(\mu_{jk}\right)_{\substack{j=1,\ldots,N \\ k=1,\ldots,K}} \right\}$$

La figure 3.9 indique la représentation graphique du modèle MNL. Ce modèle est basé sur les hypothèses suivantes :

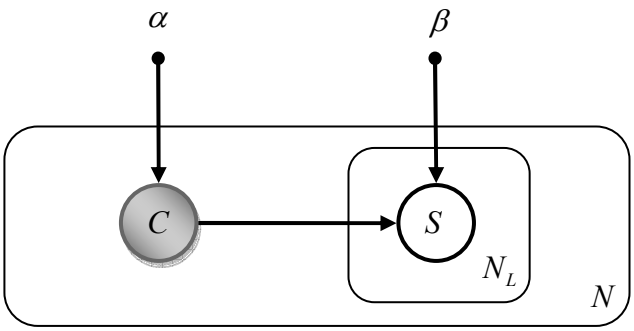


Figure 3.9- Représentation graphique du modèle MNL

i) Les liens $\{s_{ij}\}$ d'un sommet s_i sont indépendants conditionnellement à la communauté c_i de ce sommet i.e.

$$\begin{aligned} p(s_i \mid C = c_i\ ;\Theta) &= p(s_{i1},\ldots,s_{iNi} \mid C = c_i\ ;\Theta) \\ &= \text{Mult}(S\ ;N_i,\mu_{.c_i}) \\ &= \frac{N_i!}{\prod_{j=1}^{N_i} A_{ij}} \prod_{j=1}^{N_i} p(S = s_{ij} \mid C = c_i\ ;\mu_{.c_i}) \\ &= \frac{N_i!}{\prod_{j=1}^{N_i} A_{ij}} \prod_{j=1}^{N} p(S = j \mid C = c_i\ ;\mu_{.c_i})^{A_{ij}} \\ &= N_i! \prod_{j=1}^{N} \mu_{jc_i}{}^{A_{ij}} \end{aligned} \tag{3.33}$$

ii) La distribution jointe d'un nœud s_i et de sa communautés c_i est :

$$\begin{aligned} p(s_i, C = c_i\ ;\Theta) &= p(c_i\ ;\Theta)\, p(s_i \mid C = c_i\ ;\Theta) \\ &= \pi_{c_i} N_i! \prod_{j=1}^{N} \mu_{jc_i}{}^{A_{ij}} \end{aligned} \tag{3.34}$$

iii) Les sommets observés $\{s_i\}$ sont indépendants les uns des autres, d'où une log-vraisemblance du graphe observé égale à :

$$\begin{aligned} LL &= \log p((s_1,\ldots,s_N)\ ;\Theta) \\ &= \sum_{i=1}^{N} \log p(s_i\ ;\Theta) \\ &= \sum_{i=1}^{N} \log \sum_{k=1}^{K} p(s_i, C = k\ ;\Theta) \\ &= \sum_{i=1}^{N} \log \left(\sum_{k=1}^{K} \pi_k \prod_{j=1}^{N_i} \mu_{jk}{}^{A_{ij}} \right) + cst \end{aligned} \tag{3.35}$$

L'expression ci-dessus ne peut être maximisée analytiquement à cause de la présence de variables cachées dans le modèle. Il faut alors utiliser l'algorithme EM pour estimer les valeurs des paramètres Θ qui maximisent la vraisemblance du graphe.

La log-vraisemblance des données complètes est :

$$\begin{aligned} LL^C &= \log p\left((s_1,c_1),\ldots,(s_N,c_N)\ ;\Theta\right) \\ &= \sum_{i=1}^{N} \log p\left(s_i, C=c_i\ ;\Theta\right) \\ &= \sum_{i=1}^{N}\sum_{k=1}^{K} \mathbf{1}_{\{c_i=k\}} \log p\left(s_i, C=k\ ;\Theta\right) \\ &= \sum_{i=1}^{N}\sum_{k=1}^{K} \mathbf{1}_{\{c_i=k\}} \left(\log \pi_k + \sum_{j=1}^{N} A_{ij} \log \mu_{jk}\right) + cst \end{aligned} \tag{3.36}$$

La distribution a posteriori des communautés $p\left(C \mid S\ ;\Theta^{old}\right)$ est obtenue en appliquant la formule de Bayes. Pour $k=1\ldots K$ et $i=1\ldots N$, cette distribution est donnée par :

$$\begin{aligned} \omega_{ki} &= p\left(C=k \mid S=i\ ;\Theta^{old}\right) \\ &= \frac{p\left(S=i, C=k\ ;\Theta^{old}\right)}{p\left(S=i\ ;\Theta^{old}\right)} \\ &= \frac{\pi_k^{old} \prod_{j=1}^{N} \left(\mu_{jk}^{old}\right)^{A_{ij}}}{\sum_{t=1}^{K} \pi_t^{old} \prod_{j=1}^{N} \left(\mu_{jt}^{old}\right)^{A_{ij}}} \end{aligned} \tag{3.37}$$

L'espérance conditionnelle de la log-vraisemblance des données complètes est :

$$\begin{aligned} Q &= \mathbf{E}_C\left[\sum_{i=1}^{N}\sum_{k=1}^{K} \mathbf{1}_{\{c_i=k\}} \left(\log \pi_k + \sum_{j=1}^{N} A_{ij} \log \mu_{jk}\right)\right] \\ &= \sum_{i=1}^{N}\sum_{k=1}^{K} \mathbf{E}_{C_i}\left[\mathbf{1}_{\{c_i=k\}}\right] \left(\log \pi_k + \sum_{j=1}^{N} A_{ij} \log \mu_{jk}\right) \\ &= \sum_{i=1}^{N}\sum_{k=1}^{K} p\left(C=k \mid S=i\ ;\Theta^{old}\right) \left(\log \pi_k + \sum_{j=1}^{N} A_{ij} \log \mu_{jk}\right) \\ &= \sum_{i=1}^{N}\sum_{k=1}^{K} \omega_{ki} \left(\log \pi_k + \sum_{j=1}^{N} A_{ij} \log \mu_{jk}\right) \end{aligned} \tag{3.38}$$

Si l'on suppose maintenant que la distribution a posteriori des communautés est connue, les paramètres π et μ qui maximisent Q à chaque itération de l'algorithme EM sont obtenus en maximisant le lagrangien suivant :

$$H = \sum_{i=1}^{N}\sum_{k=1}^{K} \omega_{ki} \left(\log \pi_k + \sum_{j=1}^{N} A_{ij} \log \mu_{jk}\right) + \lambda\left(1 - \sum_{k=1}^{K} \pi_k\right) + \sum_{k=1}^{K} \sigma_k \left(1 - \sum_{j=1}^{N} \mu_{jk}\right) \tag{3.39}$$

où λ et $(\sigma_1,\ldots,\sigma_K)$ sont les multiplicateurs de Lagrange qui permettent de vérifier les conditions de normalisation sur les paramètres à savoir :

$$\sum_{k=1}^{K}\pi_k = 1 \quad \text{et} \quad \forall k \in \{1,\ldots,K\}, \sum_{j=1}^{N}\mu_{jk} = 1 .$$

En résolvant l'équation $\frac{\partial H}{\partial \pi_k} = 0$, on obtient l'équation de ré-estimation suivante :

$$\pi_k = \frac{1}{N}\sum_{i=1}^{N}\omega_{ki} \tag{3.40}$$

Cette expression signifie que la probabilité a priori d'une communauté *k* est proportionnelle au nombre de nœuds appartenant à cette communauté.

De même, en résolvant l'équation $\frac{\partial H}{\partial \mu_{jk}} = 0$, on obtient l'équation de ré-estimation suivante :

$$\mu_{jk} = \frac{\sum_{i=1}^{N} A_{ij}\omega_{ki}}{\sum_{i=1}^{N}\sum_{j=1}^{N} A_{ij}\omega_{ki}} \tag{3.41}$$

Les différentes étapes d'estimation des paramètres du modèle MNL sont décrites par l'algorithme 3.5.

Algorithme 3.5 : Algorithme d'estimation des paramètres du modèle MNL

Entrée : - un graphe G d'ordre N représenté par sa matrice d'adjacence $\mathbf{A}$
- le nombre de communautés K

Sortie : les paramètres du modèle à savoir le vecteur $\boldsymbol{\pi}$ et la matrice $\boldsymbol{\mu}$

début

// Initialisation

1. $\boldsymbol{\pi} \leftarrow \frac{\mathbf{1}_{K\times 1}}{K}$, $\boldsymbol{\mu} \leftarrow \frac{\mathbf{1}_{N\times K}}{N}$

// Optimisation

2. **répéter**

// Etape E de l'algorithme

3. **pour** $k = 1 \ldots K$ et $i = 1 \ldots N$ **faire**

4. $\omega_{ki} \leftarrow \frac{\pi_k \prod_{j=1}^{N} \left(\mu_{jk}\right)^{A_{ij}}}{\sum_{t=1}^{K} \pi_t \prod_{j=1}^{N} \left(\mu_{jt}\right)^{A_{ij}}}$

5. **fin**

// Etape M de l'algorithme

6. **pour** $k = 1 \ldots K$ **faire**

7. $\pi_k \leftarrow \frac{1}{N} \sum_{i=1}^{N} \omega_{ki}$

8. **pour** $j = 1 \ldots N$ **faire**

9. $\mu_{jk} \leftarrow \frac{\sum_{i=1}^{N} A_{ij} \omega_{ki}}{\sum_{i=1}^{N} \sum_{j=1}^{N} A_{ij} \omega_{ki}}$

10. **fin**

11. **fin**

12. **jusqu'à** *convergence*

fin

3.3.3 Approches basées sur le modèle PLSA (Probabilistic Latent Semantic Analysis)

PLSA est un modèle probabiliste proposé par Hofmann [Hofmann 01] pour l'analyse de données de co-occurrence. Bien qu'utilisé initialement pour la classification non supervisée de textes, le modèle PLSA a trouvé de nombreuses autres applications telles que la recherche d'information [Hofmann 99] ou les systèmes de recommandation [Popescul et al. 01]. PLSA a été employé notamment pour l'identification de structures de communautés dans les graphes ; il est en effet à la base de deux modèles d'ISC à savoir PHITS et SPAEM. Nous présentons ci-dessous chacun de ces deux modèles.

3.3.3.1 Le modèle PHITS (Probabilistic HITS)

Dans [Cohn and Chang 00], les auteurs présentent le modèle PHITS comme une alternative à l'algorithme HITS (cf. chapitre 1) pour le calcul de centralité dans les graphes de documents. Ils montrent que leur modèle est capable de trouver des ensembles d'autorités et de hubs qui peuvent être interprétés "plus facilement" que ceux calculés par l'algorithme HITS. PHITS calcule les ensembles d'autorités et de hubs en commençant d'abord par identifier la structure de communautés contenue dans le graphe de documents, puis en se basant sur cette structure, il attribue à chaque document des degrés d'autorité et d'hubité. PHITS est toutefois un modèle très peu connu dans le domaine de l'identification de structures de communautés. Nous pensons que cela est dû au fait qu'il soit présenté dans un contexte de calcul de centralité et non pas d'identification de communautés.

Il existe deux versions différentes du modèle PLSA : une version symétrique et une version asymétrique. PHITS est un modèle génératif de graphes basé sur la version asymétrique de PLSA. A la différence du modèle MNL qui est un modèle génératif des sommets, PHITS est plutôt un modèle génératif des liens entre sommets. Le modèle PHITS suppose que les M liens d'un graphe G d'ordre N représenté par sa matrice d'adjacence **A** sont générés par le processus suivant :

- Pour $m = 1$ à M faire :

1. Choisir un nœud source $x_m \sim \text{Mult}\left(1,(\pi_1,\ldots,\pi_N)\right)$ où π est un vecteur de paramètres tel que $\sum_{i=1}^{N}\pi_i = 1$.
2. Conditionnellement à x_m, choisir une communauté $c_m \sim \text{Mult}\left(1,(\mu_{1x_m},\ldots,\mu_{Kx_m})\right)$ où μ est une matrice de paramètres de dimension $K \times N$ telle que $\forall i \in \{1,\ldots,N\}, \sum_{k=1}^{K}\mu_{ki} = 1$.
3. Conditionnellement à c_m, choisir un nœud destination $y_m \sim \text{Mult}\left(1,(\phi_{1c_m},\ldots,\phi_{Nc_m})\right)$ où ϕ est une matrice de paramètres de dimension $N \times K$ telle que $\forall k \in \{1,\ldots,K\}, \sum_{j=1}^{N}\phi_{jk} = 1$.
4. Générer un lien entre le nœud x_m et le nœud y_m

Nous noterons par Θ l'ensemble des paramètres du modèle PHITS i.e. :

$$\Theta = \left\{ \left(\pi_i\right)_{i=1,\ldots,N}, \left(\mu_{ki}\right)_{\substack{k=1,\ldots,K \\ i=1,\ldots,N}}, \left(\phi_{jk}\right)_{\substack{j=1,\ldots,N \\ k=1,\ldots,K}} \right\}$$

La figure 3.10 indique la représentation graphique du modèle PHITS. Ce modèle est basé sur les hypothèses suivantes :

i) La distribution jointe d'un lien $\{(x_m, y_m)\}$ et de sa communauté c_m est :

$$\begin{aligned} p(X = x_m, Y = y_m, C = c_m \ ;\Theta) &= p(X = x_m \ ;\Theta)\, p(C = c_m \mid X = x_m \ ;\Theta) \\ &\quad p(Y = y_m \mid C = c_m \ ;\Theta) \\ &= \pi_{x_m} \mu_{c_m x_m} \phi_{y_m c_m} \end{aligned} \tag{3.42}$$

ii) Les liens observés $\{(x_m, y_m)\}$ sont indépendants les uns des autres. La log-vraisemblance du graphe observé est donc :

$$\begin{aligned} LL &= \log p\left((x_1, y_1), \ldots, (x_M, y_M) \ ;\Theta\right) \\ &= \sum_{m=1}^{M} \log p(X = x_m, Y = y_m \ ;\Theta) \\ &= \sum_{m=1}^{M} \log \sum_{k=1}^{K} p(X = x_m, Y = y_m, C = k \ ;\Theta) \\ &= \sum_{m=1}^{M} \log \sum_{k=1}^{K} \left(\pi_{x_m} \mu_{k x_m} \phi_{y_m k}\right) \end{aligned} \tag{3.43}$$

La quantité *LL* ne pouvant être maximisée directement en raison de la présence de variables latentes, il est nécessaire de recourir à l'algorithme EM pour l'estimation des paramètres de ce modèle.

La log-vraisemblance des données complètes (i.e. celle des triplets $\{(x_m, y_m, c_m)\}$) est :

$$\begin{aligned} LL^C &= \log\left(p\left((x_1, y_1, c_1), \ldots, (x_M, y_M, c_M) \ ;\Theta\right)\right) \\ &= \sum_{m=1}^{M} \log p(X = x_m, Y = y_m, C = c_m \ ;\Theta) \\ &= \sum_{m=1}^{M} \sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{k=1}^{K} \mathbf{1}_{\{x_m = i, y_m = j, c_m = k\}} \left(\log \pi_i + \log \mu_{ki} + \log \phi_{jk}\right) \end{aligned} \tag{3.44}$$

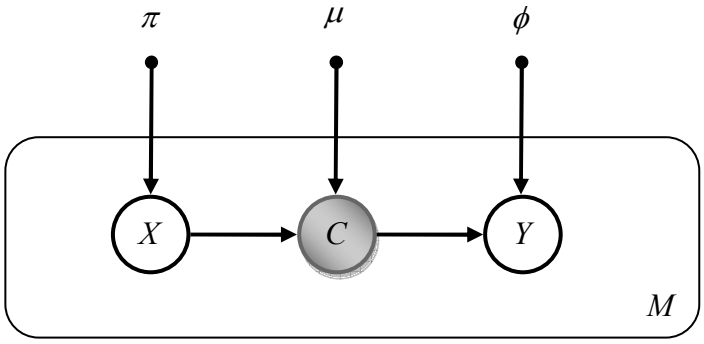


Figure 3.10 - Représentation graphique du modèle PHITS

La distribution a posteriori des communautés est obtenue en appliquant la formule de Bayes. Pour $i=1\ldots N,\ j=1\ldots N,\ k=1\ldots K$, cette distribution est donnée par :

$$\begin{aligned}\omega_{kij} &= p\left(C=k \mid X=i, Y=j\ ;\Theta^{old}\right)\\ &= \frac{p\left(X=i, Y=j, C=k\ ;\Theta^{old}\right)}{p\left(X=i, Y=j\ ;\Theta^{old}\right)}\\ &= \frac{\mu_{ki}^{old}\,\phi_{jk}^{old}}{\sum_{t=1}^{K}\mu_{ti}^{old}\,\phi_{jt}^{old}}\end{aligned} \tag{3.45}$$

L'espérance conditionnelle de la log-vraisemblance des données complètes est donc :

$$\begin{aligned}Q &= \mathbf{E}_C\left[\sum_{m=1}^{M}\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{K}\mathbf{1}_{\{x_m=i,y_m=j,c_m=k\}}\left(\log\pi_i+\log\mu_{ki}+\log\phi_{jk}\right)\right]\\ &= \sum_{m=1}^{M}\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{K}\mathbf{1}_{\{x_m=i,y_m=j\}}\mathbf{E}_{C_m}\left[\mathbf{1}_{\{c_m=k\}}\right]\left(\log\pi_i+\log\mu_{ki}+\log\phi_{jk}\right)\\ &= \sum_{i=1}^{N}\sum_{j=1}^{N}A_{ij}\log\pi_i+\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{K}A_{ij}\omega_{kij}\left(\log\mu_{ki}+\log\phi_{jk}\right)\end{aligned} \tag{3.46}$$

Les nouveaux paramètres Θ qui maximisent cette espérance sont obtenus en maximisant le lagrangien suivant :

$$\begin{aligned}H &= \sum_{i=1}^{N}\sum_{j=1}^{N}A_{ij}\log\pi_i+\sum_{i=1}^{N}\sum_{j=1}^{N}A_{ij}\sum_{k=1}^{K}\omega_{kij}\left(\log\mu_{ki}+\log\phi_{jk}\right)+\lambda\left(1-\sum_{i=1}^{N}\pi_i\right)\\ &+\sum_{i=1}^{N}\sigma_i\left(1-\sum_{k=1}^{K}\mu_{ki}\right)+\sum_{k=1}^{K}\zeta_k\left(1-\sum_{j=1}^{N}\phi_{jk}\right)\end{aligned} \tag{3.47}$$

où λ, $(\sigma_1,\ldots,\sigma_N)$ et $(\zeta_1,\ldots,\zeta_K)$ sont les multiplicateurs de Lagrange.

En résolvant l'équation $\frac{\partial H}{\partial \pi_i}=0$, on obtient l'équation de ré-estimation suivante:

$$\pi_i = \frac{1}{N}\sum_{j=1}^{N}A_{ij} \tag{3.48}$$

Cela signifie que la probabilité a priori d'un nœud *i* est proportionnelle au nombre de liens sortants que possède ce nœud.

Les équations de ré-estimation de μ_{ki} et ϕ_{jk} sont obtenues en résolvant respectivement les équations $\frac{\partial H}{\partial \mu_{ki}}=0$ et $\frac{\partial H}{\partial \phi_{jk}}=0$:

$$\mu_{ki} = \frac{\sum_{j=1}^{N}A_{ij}\omega_{kij}}{\sum_{j=1}^{N}A_{ij}} \tag{3.49}$$

$$\phi_{jk} = \frac{\sum_{i=1}^{N} A_{ij}\omega_{kij}}{\sum_{i=1}^{N}\sum_{t=1}^{N} A_{ij}\omega_{kit}} \tag{3.50}$$

L'algorithme EM complet pour l'estimation des paramètres du modèle PHITS est décrit par l'algorithme 3.6.

Algorithme 3.6 : Algorithme d'estimation des paramètres du modèle PHITS

Entrée : - un graphe G d'ordre N représenté par sa matrice d'adjacence $\mathbf{A}$
- le nombre de communautés K

Sortie : les paramètres du modèle à savoir le vecteur $\boldsymbol{\pi}$ et les matrices $\boldsymbol{\mu}$ et $\boldsymbol{\varphi}$

début

// Initialisation

1. $\boldsymbol{\mu} \leftarrow \frac{\mathbf{1}_{K\times N}}{K}$, $\boldsymbol{\varphi} \leftarrow \frac{\mathbf{1}_{N\times K}}{N}$

// Optimisation

2. **pour** $i = 1 \ldots N$ **faire**
3. $\quad \pi_i = \frac{1}{N}\sum_{j=1}^{N} A_{ij}$
4. **fin**
5. **répéter**
   // Etape E de l'algorithme
6. $\quad$ **pour** $i = 1 \ldots N$, $j = 1 \ldots N$ et $k = 1 \ldots K$ **faire**
7. $\quad\quad \omega_{kij} \leftarrow \frac{\mu_{ki}\phi_{jk}}{\sum_{t=1}^{K}\mu_{ti}\phi_{jt}}$
8. $\quad$ **fin**

   // Etape M de l'algorithme
9. $\quad$ **pour** $i = 1 \ldots N$ et $k = 1 \ldots K$ **faire**
10. $\quad\quad \mu_{ki} = \frac{\sum_{j=1}^{N} A_{ij}\omega_{kij}}{\sum_{j=1}^{N} A_{ij}}$
11. $\quad\quad \phi_{ik} = \frac{\sum_{l=1}^{N} A_{li}\omega_{kli}}{\sum_{l=1}^{N}\sum_{t=1}^{N} A_{li}\omega_{klt}}$
12. $\quad$ **fin**
13. **jusqu'à** *convergence*

fin

3.3.3.2 Le modèle SPAEM

En se basant sur la version symétrique du modèle PLSA, Ren et al. [Ren et al. 09] ont proposé récemment le modèle SPAEM pour l'identification de structures de communautés dans les graphes non-orientés. Le modèle SPAEM suppose que les *M* arêtes d'un graphe G d'ordre *N* représenté par sa matrice d'adjacence **A** sont générées par le processus suivant :

- Pour $m = 1$ à M faire :

1. Choisir une communauté $c_m \sim \text{Mult}\left(1,(\pi_1,\ldots,\pi_K)\right)$ où π est un vecteur de paramètres tel que $\sum_{k=1}^{K}\pi_k = 1$.
2. Conditionnellement à c_m, choisir un nœud $x_m \sim \text{Mult}\left(1,(\mu_{1c_m},\ldots,\mu_{Nc_m})\right)$ où μ est une matrice de paramètres de dimension $N \times K$ telle que $\forall k \in \{1,\ldots,K\}, \sum_{i=1}^{N}\mu_{ik} = 1$.
3. Conditionnellement à c_m, choisir un sommet (ou nœud) $y_m \sim \text{Mult}\left(1,(\mu_{1c_m},\ldots,\mu_{Nc_m})\right)$
4. Générer une arête entre les nœuds x_m et y_m.

Nous noterons par Θ l'ensemble des paramètres du modèle SPAEM i.e. :

$$\Theta = \left\{ \left(\pi_k\right)_{k=1,\ldots,K}, \left(\mu_{ik}\right)_{\substack{i=1,\ldots,N \\ k=1,\ldots,K}} \right\}$$

La représentation graphique du modèle SPAEM est indiquée par la figure 3.11. Ce modèle est basé sur les hypothèses suivantes :

i) La distribution jointe d'une arête $\{(x_m, y_m)\}$ et de sa communauté c_m est :

$$\begin{aligned} p\left(X = x_m, Y = y_m, C = c_m \;;\Theta\right) &= p\left(C = c_m \;;\Theta\right) p\left(X = x_m \mid C = c_m \;;\Theta\right) p\left(Y = y_m \mid C = c_m \;;\Theta\right) \\ &= \pi_{c_m} \mu_{x_m c_m} \mu_{y_m c_m} \end{aligned} \quad (3.51)$$

ii) Les arêtes observées $\{(x_m, y_m)\}$ sont indépendantes les unes des autres. La log-vraisemblance du graphe observé est donc :

$$\begin{aligned} LL &= \log p\left((x_1,y_1),\ldots,(x_M,y_M) \;;\Theta\right) \\ &= \sum_{m=1}^{M} \log p\left(X = x_m, Y = y_m \;;\Theta\right) \\ &= \sum_{m=1}^{M} \log \sum_{k=1}^{K} p\left(X = x_m, Y = y_m, C = k \;;\Theta\right) \\ &= \sum_{m=1}^{M} \log \sum_{k=1}^{K} \left(\pi_k \mu_{x_m k} \mu_{y_m k}\right) \end{aligned} \quad (3.52)$$

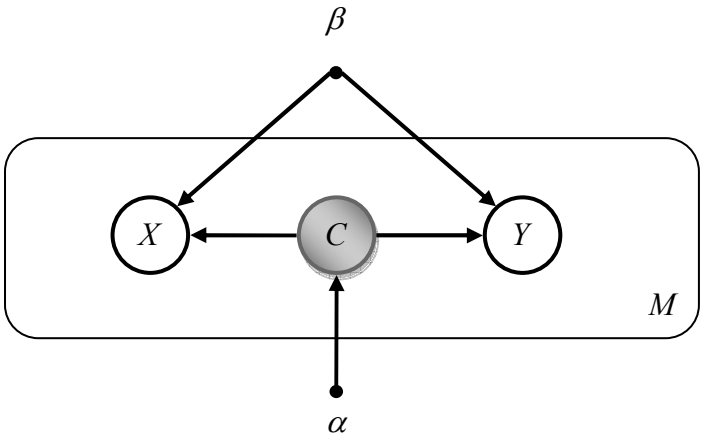


Figure 3.11 - Représentation graphique du modèle SPAEM

Comme pour l'algorithme PHITS, l'estimation des paramètres du modèle SPAEM doit être réalisée en utilisant l'algorithme EM. La log-vraisemblance des données complètes est :

$$\begin{aligned} LL^C &= \log p\left(\left(x_1, y_1, c_1\right), \ldots, \left(x_M, y_M, c_M\right) \ ; \Theta\right) \\ &= \sum_{m=1}^{M} \log p\left(X = x_m, Y = y_m, C = c_m \ ; \Theta\right) \\ &= \sum_{m=1}^{M} \sum_{i=1}^{N} \sum_{j=i+1}^{N} \sum_{k=1}^{K} \mathbf{1}_{\{x_m = i, y_m = j, c_m = k\}} \log p\left(X = i, Y = j, C = k \ ; \Theta\right) \\ &= \sum_{m=1}^{M} \sum_{i=1}^{N} \sum_{j=i+1}^{N} \sum_{k=1}^{K} \mathbf{1}_{\{x_m = i, y_m = j, c_m = k\}} \left(\log \pi_k + \log \mu_{ik} + \log \mu_{jk}\right) \end{aligned} \tag{3.53}$$

La distribution a posteriori des variables latentes (i.e. des communautés) est obtenue en appliquant la formule de Bayes. Pour $i = 1 \ldots N$, $j = 1 \ldots N$, $k = 1 \ldots K$, cette distribution est donnée par :

$$\begin{aligned} \omega_{kij} &= p\left(C = k \mid X = i, Y = j \ ; \Theta^{old}\right) \\ &= \frac{p\left(X = i, Y = j, C = k \ ; \Theta^{old}\right)}{p\left(X = i, Y = j \ ; \Theta^{old}\right)} \\ &= \frac{\pi_k \mu_{ik} \mu_{jk}}{\sum_{t=1}^{K} \pi_t \mu_{it} \mu_{jt}} \end{aligned} \tag{3.54}$$

L'espérance de la log-vraisemblance des données complètes, conditionnellement aux communautés, est donc :

$$
\begin{aligned}
Q &= \mathbf{E}_C\left[\sum_{m=1}^{M}\sum_{i=1}^{N}\sum_{j=i+1}^{N}\sum_{k=1}^{K}\mathbf{1}_{\{x_m=i,y_m=j,c_m=k\}}\left(\log\pi_k+\log\mu_{ik}+\log\mu_{jk}\right)\right] \\
&= \sum_{m=1}^{M}\sum_{i=1}^{N}\sum_{j=i+1}^{N}\sum_{k=1}^{K}\mathbf{1}_{\{x_m=i,y_m=j\}}\mathbf{E}_{C_m}\left[\mathbf{1}_{\{c_m=k\}}\right]\left(\log\pi_k+\log\mu_{ik}+\log\mu_{jk}\right) \\
&= \sum_{m=1}^{M}\sum_{i=1}^{N}\sum_{j=i+1}^{N}\sum_{k=1}^{K}\mathbf{1}_{\{x_m=i,y_m=j\}}\,p\left(C=k\mid X=i,Y=j\;;\Theta^{old}\right)\left(\log\pi_k+\log\mu_{ik}+\log\mu_{jk}\right) \\
&= \sum_{i=1}^{N}\sum_{j=i+1}^{N}\sum_{k=1}^{K}A_{ij}\omega_{kij}\left(\log\pi_k+\log\mu_{ik}+\log\mu_{jk}\right)
\end{aligned}
\tag{3.55}
$$

La somme sur la variable j commence par la valeur $i+1$ pour ne pas que les liens soient considérés deux fois ; la matrice d'adjacence **A** étant symétrique, nous avons : $A_{ij}=A_{ji}$. De plus la variable j commence par la valeur $i+1$ et non pas i car le modèle suppose que le graphe observé ne contient pas de boucles.

Le lagrangien à maximiser pour obtenir nouvelles valeurs des paramètres Θ est :

$$
\begin{aligned}
H = &\sum_{i=1}^{N}\sum_{j=i+1}^{N}\sum_{k=1}^{K}A_{ij}\omega_{kij}\left(\log\pi_k+\log\mu_{ik}+\log\mu_{jk}\right) \\
&+\lambda\left(1-\sum_{k=1}^{K}\pi_i\right)+\sum_{k=1}^{K}\sigma_k\left(1-\sum_{i=1}^{N}\mu_{ik}\right)
\end{aligned}
\tag{3.56}
$$

où λ et $\left(\sigma_1,\ldots,\sigma_K\right)$ sont les multiplicateurs de Lagrange.

En résolvant l'équation $\frac{\partial H}{\partial \pi_k}=0$, on obtient l'équation de ré-estimation suivante :

$$
\pi_k=\frac{1}{N}\sum_{i=1}^{N}\sum_{j=i+1}^{N}A_{ij}\omega_{kij} \tag{3.57}
$$

Cela signifie que la probabilité a priori d'une communauté *k* est proportionnelle au nombre de nœuds appartenant à cette communauté.

L'équation de ré-estimation de μ_{ik} obtenue en résolvant l'équation $\frac{\partial H}{\partial \mu_{ik}}=0$ est :

$$
\mu_{ik}=\frac{\sum_{j=i+1}^{N}A_{ij}\omega_{kij}}{\sum_{l=1}^{N}\sum_{j=l+1}^{N}A_{ij}\omega_{klj}} \tag{3.58}
$$

L'algorithme 3.7 décrit les différentes étapes d'estimation des paramètres du modèle SPAEM.

Algorithme 3.7 : Algorithme d'estimation des paramètres du modèle SPAEM

Entrée : - un graphe non-orienté G d'ordre N représenté par sa matrice d'adjacence $\mathbf{A}$
- le nombre de communautés K

Sortie : les paramètres du modèle à savoir le vecteur $\boldsymbol{\pi}$ et la matrice $\boldsymbol{\mu}$

début

// Initialisation

1. $\boldsymbol{\pi} \leftarrow \frac{\mathbf{1}_{K\times 1}}{K}$, $\boldsymbol{\mu} \leftarrow \frac{\mathbf{1}_{N\times K}}{N}$

// Optimisation

2. **répéter**

// Etape E de l'algorithme

3. **pour** $i = 1 \ldots N$, $j = 1 \ldots N$ et $k = 1 \ldots K$ **faire**

4. $\omega_{kij} \leftarrow \frac{\pi_k \mu_{ik} \mu_{jk}}{\sum_{t=1}^{K} \pi_t \mu_{it} \mu_{jt}}$

5. **fin**

// Etape M de l'algorithme

6. **pour** $k = 1 \ldots K$ **faire**

7. $\pi_k = \frac{1}{N} \sum_{i=1}^{N} \sum_{j=i+1}^{N} A_{ij} \omega_{kij}$

8. **pour** $i = 1 \ldots N$ **faire**

9. $\mu_{ik} = \frac{\sum_{j=i+1}^{N} A_{ij} \omega_{kij}}{\sum_{l=1}^{N} \sum_{j=l+1}^{N} A_{ij} \omega_{klj}}$

10. **fin**

11. **fin**

12. **jusqu'à** *convergence*

fin

3.3.4 Approches basées sur le modèle SBM (Stochastic BlockModels)

Les SBM sont les modèles génératifs les plus connus dans le domaine de l'identification de structures de communautés dans les réseaux sociaux. Ils ont été introduits comme une extension des modèles en blocs (Block Models) classiques [Anderson et al. 92][Wasserman and Faust 94]. Ces derniers ont pour but d'identifier dans un réseau social les groupes d'acteurs possédant les mêmes caractéristiques. Ils sont basés sur le principe de *l'équivalence structurelle* entre deux acteurs : deux acteurs i et j sont dits structurellement équivalents si et seulement si ils ont le même voisinage immédiat i.e. ils interagissent avec les mêmes acteurs. Cette définition de l'équivalence entre deux acteurs est assez contraignante puisqu'il suffit qu'il y ait une seule différence entre les connexions des nœuds pour que les deux nœuds ne soient plus équivalents. Or, en pratique, deux nœuds peuvent être équivalents sans avoir exactement les mêmes liens. Pour prendre en compte cette idée, Holland et al. ont proposé la

notion d'*équivalence stochastique* qui suppose que deux nœuds peuvent être équivalents même s'ils n'ont pas exactement les mêmes liens. Il s'agit en fait d'une équivalence beaucoup plus souple et qui peut par conséquent être utilisée avec des réseaux réels.

Novicky et Snijders [Nowicki and Snijders 01] ont proposé un SBM général permettant d'identifier des communautés dans différents types de graphes (orientés/non-orientés, signés/non-signés, graphes simples/multigraphes). Nous présentons ici une version de ce SBM pour des graphes simples, orientés et sans boucles.

Le modèle SBM suppose que les liens d'un graphe G d'ordre *N* représenté par sa matrice d'adjacence **A** sont générés par le processus suivant :

1. Pour chaque sommet i, $i = 1,\ldots,N$ faire :

   i. Choisir une communauté $c_i \sim \text{Mult}\left(1,(\pi_1,\ldots,\pi_K)\right)$ où π est un vecteur de paramètres tel que $\sum_{k=1}^{K}\pi_k = 1$.

2. Pour chaque couple de sommets (i,j), $i = 1,\ldots,N$, $j = 1,\ldots,N$ faire :

   i. Conditionnellement à c_i et c_j, tirer une relation $r_{ij} \sim \text{Bern}\left(r_{ij}\,;\mu_{c_i c_j}\right)$ où **μ** est une matrice de paramètres de dimension $K \times K$, appelée aussi matrice des blocs.

   ii. Si $r_{ij} = 1$ alors générer un lien entre le nœud *i* et le nœud *j*.

Nous noterons par Θ l'ensemble des paramètres du modèle SBM i.e. :

$$\Theta = \left\{ \left(\pi_k\right)_{k=1,\ldots,K}, \left(\mu_{jl}\right)_{\substack{j=1,\ldots,K \\ l=1,\ldots,K}} \right\}$$

La figure 3.12 indique la représentation graphique du modèle SBM. Ce modèle est basé sur les hypothèses suivantes :

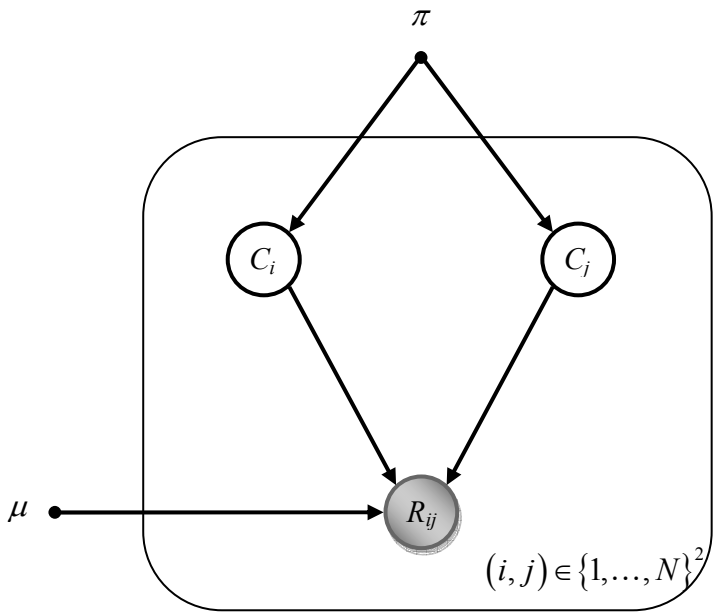


Figure 3.12 - Représentation graphique du modèle SBM

i) Les communautés $\{c_i\}$ des sommets sont indépendantes les unes des autres i.e.

$$\begin{aligned} p(C\ ;\Theta) &= p\left((c_1,\ldots,c_N)\ ;\Theta\right) \\ &= \prod_{i=1}^{N} p\left(C = c_i\ ;\Theta\right) \\ &= \prod_{i=1}^{N} \pi_{c_i} \end{aligned} \tag{3.59}$$

ii) Les relations (présence ou absence de lien) $\{r_{ij}\}$ entre les nœuds sont indépendantes les unes des autres lorsque la communauté de chaque nœud est connue i.e.

$$\begin{aligned} p(R \mid C\ ;\Theta) &= p\left((r_{11}, r_{12},\ldots, r_{21},\ldots, r_{NN}) \mid (c_1,\ldots,c_n)\ ;\Theta\right) \\ &= \prod_{i=1}^{N}\prod_{j\neq i}^{N} p\left(R = r_{ij} \mid C_i = c_i, C_j = c_j\ ;\Theta\right) \\ &= \prod_{i=1}^{N}\prod_{j\neq i}^{N} \mathrm{Bern}\left(A_{ij}\ ;\mu_{c_i c_j}\right) \\ &= \prod_{i=1}^{N}\prod_{j\neq i}^{N} \mu_{c_i c_j}{}^{A_{ij}} \left(1-\mu_{c_i c_j}\right)^{1-A_{ij}} \end{aligned} \tag{3.60}$$

La vraisemblance des relations observées est :

$$\begin{aligned} L &= p(R\ ;\Theta) \\ &= \sum_{C} p(R, C\ ;\Theta) \\ &= \sum_{k_1=1}^{K}\cdots\sum_{k_N=1}^{K} p\left((r_{11}, r_{12},\ldots, r_{21},\ldots, r_{NN}), (C_1 = k_1,\ldots, C_N = k_N)\ ;\Theta\right) \end{aligned} \tag{3.61}$$

Cette quantité ne peut être calculée ou maximisée analytiquement car elle contient une somme de K^N éléments. De plus, l'algorithme EM ne peut pas être utilisé ici car la distribution a posteriori des variables cachées (i.e. des communautés) ne peut non plus être calculée. En effet, celle-ci est donnée par :

$$P(C \mid R\ ;\Theta) = P(C_1,\ldots, C_N \mid R\ ;\Theta)$$

et ne peut être factorisée en un produit de distributions indépendantes comme c'est le cas pour les modèles MNL et PLSA. Cela est dû au fait que les variables cachées $\{C_i\}$ ne sont plus indépendantes lorsque les variables $\{R_{ij}\}$ sont observées. Si l'on considère par exemple le cas d'une relation R_{ij} et de deux variables cachées C_i et C_j, la probabilité a postériori des variables C_i et C_j est :

$$
\begin{aligned}
P\left(C_i, C_j \mid R_{ij}\right) &= \frac{P\left(C_i, C_j, R_{ij}\right)}{P\left(R_{ij}\right)} \\
&= \frac{P\left(C_i\right) P\left(C_j\right) P\left(R_{ij} \mid C_i, C_j\right)}{P\left(R_{ij}\right)} \\
&\neq P\left(C_i \mid R_{ij}\right) P\left(C_j \mid R_{ij}\right)
\end{aligned}
$$

Les deux variables C_i et C_j ne sont donc pas indépendantes conditionnellement à R_{ij} puisque leur distribution a postériori ne peut être factorisée.

Nous allons décrire ici l'utilisation de l'inférence variationnelle employée notamment par Daudin et al. [Daudin et al. 08] pour l'estimation des paramètres du modèle SBM. Rappelons que le principe de l'inférence variationnelle est de chercher une distribution paramétrique q qui possède une forme simple, et qui soit une bonne approximation de la distribution a posteriori des variables cachées. Pour le modèle SBM, la distribution q est choisie telle que :

$$
q\left(C\right) = q\left(C_1, \ldots, C_N\right) = \prod_{i=1}^{N} q\left(C_i\right) \tag{3.62}
$$

Cela revient à faire l'hypothèse que les variables $\{C_i\}$ sont indépendantes. Une fois la forme de la distribution q choisie, l'étape suivante est de déterminer les paramètres de chacune des distributions $q(C_i)$. D'après la théorie de l'inférence variationnelle, nous avons [Bishop 07] :

$$
\begin{aligned}
\log q\left(C_i\right) &= \mathbf{E}_{C\setminus^i}\left[\log p\left(R, C\ ; \Theta\right)\right] + cst \\
&= \mathbf{E}_{C\setminus^i}\left[\log p\left(R \mid C\ ; \Theta\right) + \log p\left(C\ ; \Theta\right)\right] + cst \\
&= \mathbf{E}_{C\setminus^i}\left[\sum_{i=1}^{N}\sum_{j\neq i}^{N} \log\left(\mu_{c_i c_j}^{A_{ij}}\left(1-\mu_{c_i c_j}\right)^{1-A_{ij}}\right) + \sum_{i=1}^{N} \log \pi_{c_i}\right] + cst \\
&= \mathbf{E}_{C\setminus^i}\left[\sum_{i=1}^{N}\sum_{j\neq i}^{N}\sum_{k=1}^{K}\sum_{l=1}^{K} \mathbf{1}_{\{c_i=k, c_j=l\}} \log\left(\mu_{kl}^{A_{ij}}\left(1-\mu_{kl}\right)^{1-A_{ij}}\right) + \sum_{i=1}^{N}\sum_{k=1}^{K} \mathbf{1}_{\{c_i=k\}} \log \pi_k\right] + cst \\
&= 2\times\sum_{j\neq i}^{N}\sum_{k=1}^{K}\sum_{l=1}^{K} \mathbf{1}_{\{c_i=k\}} \mathbf{E}_{C_j}\left[\mathbf{1}_{\{c_j=l\}}\right] \log\left(\mu_{kl}^{A_{ij}}\left(1-\mu_{kl}\right)^{1-A_{ij}}\right) + \sum_{k=1}^{K} \mathbf{1}_{\{c_i=k\}} \log \pi_k + cst \\
&= \sum_{k=1}^{K} \mathbf{1}_{\{c_i=k\}} \left(2\times\sum_{j\neq i}^{N}\sum_{l=1}^{K} \omega_{jl} \log\left(\mu_{kl}^{A_{ij}}\left(1-\mu_{kl}\right)^{1-A_{ij}}\right) + \log \pi_k\right) + cst
\end{aligned} \tag{3.63}
$$

où $C\setminus^i$ désigne l'ensemble des variables C moins la variable C_i, et $\omega_{jl} = \mathbf{E}_{C_j}\left[\mathbf{1}_{\{c_j=l\}}\right]$ correspond à la probabilité a postériori que le nœud j appartienne à la communauté l. En posant :

$$
\log \omega_{ik} = 2\times\sum_{j\neq i}^{N}\sum_{l=1}^{K} \omega_{jl} \log\left(\mu_{kl}^{A_{ij}}\left(1-\mu_{kl}\right)^{1-A_{ij}}\right) + \log \pi_k \tag{3.64}
$$

on obtient :

$$q(C_i) \propto \prod_{k=1}^{K} \omega_{ik}^{\mathbf{1}_{\{c_i=k\}}} \tag{3.65}$$

On remarque donc que la distribution variationnelle $q(C_i)$ correspond à une distribution multinomiale dont les paramètres sont donnés par l'équation du point fixe suivante, pour $k = 1 \ldots K$:

$$\omega_{ik} \propto \pi_k \prod_{j \neq i}^{N} \prod_{l=1}^{K} \left(\mu_{kl}^{A_{ij}} \left(1 - \mu_{kl}\right)^{\left(1 - A_{ij}\right)} \right)^{2 \times \omega_{jl}} \tag{3.66}$$

La distribution a posteriori de toutes les variables cachées est obtenue en appliquant de manière récursive l'équation 1.58 pour $i = 1 \ldots N$ jusqu'à la convergence de cette distribution.

En utilisant la technique des multiplicateurs de Lagrange pour maximiser l'espérance de la log-vraisemblance des données complètes, on obtient les équations de ré-estimation suivantes, pour $k = 1 \ldots K$, $l = 1 \ldots K$:

$$\pi_k = \frac{1}{N} \sum_{i=1}^{N} \omega_{ik} \tag{3.67}$$

$$\mu_{kl} = \frac{\sum_{i=1}^{N} \sum_{j=1}^{N} \omega_{ik} \omega_{jl} R_{ij}}{\sum_{i=1}^{N} \sum_{j=1}^{N} \omega_{ik} \omega_{jl}} \tag{3.68}$$

L'algorithme d'estimation des paramètres du modèle SBM est donné par l'algorithme 3.8.

Dans la littérature sur l'identification de communautés, il existe plusieurs variantes du modèle SBM général que nous avons présenté dans cette section. Par exemple, Hastings [Hastings 06] a proposé un SBM qui suppose l'existence de deux types de liens entre les nœuds : soit des liens entre des nœuds de la même communauté, soit des liens entre des nœuds appartenant à des communautés différentes. Concrètement, il s'agit d'un SBM pour lequel la matrice des blocs $\boldsymbol{\pi}$ est telle que :

$$\pi_{ij} = \begin{cases} P^{in} & \text{si } i = j \\ P^{out} & \text{sinon} \end{cases}$$

où $P^{in}, P^{out} \in [0,1]$ sont des probabilités indiquant respectivement la probabilité d'un lien entre deux nœuds de la même communauté et la probabilité d'un lien entre deux nœuds appartenant à des communautés différentes. Ces deux probabilités ne sont toutefois pas estimées par le modèle mais doivent plutôt être précisées par l'utilisateur.

Hofman et Wiggins [Hofman and Wiggins 08] ont proposé un modèle équivalent à celui de Hastings sauf que les paramètres P^{in} et P^{out} sont estimés à partir des données. Latouche et al. [Latouche et al. 10] ont proposé une version bayésienne du modèle SBM que nous avons présenté ici. Pour l'estimation des paramètres de leur modèle bayésien, ils ont utilisé l'inférence variationnelle.

Algorithme 3.8 : Algorithme d'estimation des paramètres du modèle SBM

Entrée : - un graphe G d'ordre N représenté par sa matrice d'adjacence $\mathbf{A}$
- le nombre de communautés K

Sortie : les paramètres du modèle à savoir le vecteur $\boldsymbol{\pi}$ et la matrice $\boldsymbol{\mu}$

début

// Initialisation

1. $\boldsymbol{\omega} \leftarrow \frac{\mathbf{1}_{N\times K}}{K}$, $\boldsymbol{\pi} \leftarrow \frac{\mathbf{1}_{K\times 1}}{K}$, $\boldsymbol{\mu} \leftarrow rand\left(K,K\right)$

// Optimisation

2. **répéter**

// Etape E de l'algorithme

3. **pour** $i = 1,\ldots,N$ **faire**
4. **pour** $k = 1,\ldots,K$ **faire**
5. $\omega_{ik} \leftarrow \pi_k \prod_{j\neq i}^{N}\prod_{l=1}^{K}\left(\mu_{kl}^{\,R_{ij}}\left(1-\mu_{kl}\right)^{\left(1-R_{ij}\right)}\right)^{2\times\omega_{jl}}$
6. **fin**
7. $\boldsymbol{\omega}_{i.} \leftarrow \frac{\boldsymbol{\omega}_{i.}}{\left\|\boldsymbol{\omega}_{i.}\right\|_1}$
8. **fin**

// Etape M de l'algorithme

9. **pour** $k = 1,\ldots,K$ **faire**
10. $\pi_k = \frac{1}{N}\sum_{i=1}^{N}\omega_{ik}$
11. **pour** $l = 1,\ldots,K$ **faire**
12. $\mu_{kl} = \frac{\sum_{i=1}^{N}\sum_{j=1}^{N}\omega_{ik}\omega_{jl}A_{ij}}{\sum_{i=1}^{N}\sum_{j=1}^{N}\omega_{ik}\omega_{jl}}$
13. **fin**
14. **fin**
15. **jusqu'à** *convergence*

fin

3.4 Synthèse des différentes approches présentées et leur adéquation à l'analyse des graphes de documents

Dans le premier chapitre de cette thèse, nous avons mis l'accent sur le fait que les graphes de documents possèdent certaines caractéristiques qui rendent la tâche d'identification de communautés plus complexe que pour d'autres types de graphes. Dans cette section, nous étudions l'adéquation des différents algorithmes présentés pour l'identification de communautés dans les graphes de documents. Plus précisément, nous analysons chacune des approches présentées en utilisant les critères suivants :

1) Identification de communautés dans des graphes orientés : le premier critère et sans doute le plus important est l'aptitude de l'algorithme à identifier des communautés dans des graphes orientés. En effet, dans le cas des graphes de documents, l'orientation des liens possède une sémantique importante qu'il serait incorrect de négliger ou d'ignorer.

2) Identification de communautés qui se recouvrent : une autre particularité des graphes de documents est qu'ils contiennent des communautés qui se recouvrent (i.e. qui ne sont pas disjointes). Ce recouvrement est dû au fait qu'un document peut traiter plusieurs thématiques à la fois et peut ainsi appartenir à plusieurs communautés en même temps. A titre d'exemple, prenons un document en bioinformatique qui traite des données en biologie et qui utilise des techniques d'analyse issues de l'informatique. Le document fera donc référence (au moins) à des travaux en biologie et en informatique. Des approches d'ISC ne prenant pas en compte le recouvrement assigneront un tel document à une des deux communautés (biologie ou informatique) ce qui ne reflète pas de manière fidèle la réalité. En résumé, cette propriété indique si l'algorithme d'ISC prend ou ne prend pas en compte le chevauchement des communautés.

3) Connaissances a priori : il s'agit d'informations dont a besoin l'algorithme pour pouvoir s'exécuter. Ces connaissances a priori peuvent par exemple correspondre au nombre ou à la taille des communautés ou encore à des paramètres spécifiques à l'algorithme utilisé ; l'algorithme idéal pour l'ISC étant sans doute celui qui ne nécessite aucune information a priori sur la structure de communautés à identifier.

4) Complexité : les graphes de documents pouvant atteindre des tailles de l'ordre de plusieurs millions de nœuds (voire beaucoup plus), il est nécessaire que l'algorithme d'ISC soit capable d'analyser en un temps "raisonnable" ce type de graphes.

5) Déterministe : ce critère indique si le résultat final de l'algorithme dépend ou non de l'étape d'initialisation. Un algorithme déterministe trouve toujours la même structure de communautés quelque soient les conditions initiales d'exécution.

6) Identification de communautés de tailles et/ou de densités différentes : cette propriété fournit une indication sur l'aptitude de l'algorithme à identifier des communautés ayant des caractéristiques différentes (i.e. des communautés hétérogènes). Un bon algorithme d'ISC doit être capable de détecter différents types de communautés sans faire d'hypothèses a priori sur leur taille ou leur forme.

7) Multi-vue sur la structure de communautés : les sommets d'un graphe orienté peuvent être regroupés en communautés de deux façons : soit en utilisant l'information sur les liens entrants, soit en utilisant l'information sur les liens sortants. Pour les graphes de documents, il est intéressant par exemple de connaître la (les) communauté(s) d'un document par rapport aux documents qui le citent ainsi que sa (ses) communauté(s) par rapport aux documents qu'il cite ; ces deux communautés n'étant pas forcément les mêmes. Un document peut en effet citer principalement les documents d'une certaine communauté (ou thématique) alors qu'il est lui-même cité par un grand nombre d'articles appartenant à une communauté différente. Nous verrons dans le chapitre suivant que cette propriété permet notamment d'identifier les membres les plus représentatifs d'une communauté.

Les tableaux 3.1 et 3.2 résument les propriétés des différents algorithmes d'ISC présentés dans ce chapitre. Nous commentons ci-dessous les informations reportées sur ces deux tableaux.

Le principal avantage de l'algorithme des K-moyennes est sa rapidité. Cependant, la qualité de la structure de communautés qu'il identifie dépend des centroïdes initiaux. Un "mauvais" choix de ces centroîdes conduit à une structure de communautés de faible qualité. L'algorithme des K-moyennes peut identifier des communautés de tailles et de densités différentes à condition que les communautés soient suffisamment séparées dans l'espace métrique. En effet, les clusters (i.e. communautés) calculés par cet algorithme ont une forme hyper-sphérique ce qui peut causer la fusion des clusters proches.

La complexité des algorithmes de Newman & Girvan et de Donetti & Munoz est quadratique par rapport au nombre de liens pour le premier et par rapport au nombre de sommets pour le deuxième. Leur utilisation est donc restreinte à des graphes de taille limitée. Le fait que ces algorithmes soient basés sur la modularité de Newman les rend incapables d'identifier des communautés dont la taille est inférieure à un certain seuil (cf. problème de la modularité dans la section 3.1.3). Il est par ailleurs nécessaire de préciser pour l'algorithme de Donetti et Munoz le nombre de vecteurs propres qui formeront l'espace de projection.

Propriété	**KM**	**GN**	**DM**	**BS**
Identification de communautés dans des graphes orientés	oui	non	oui	non
Identification de communautés qui se recouvrent	non	non	non	non
Connaissances a priori	Nombre de communautés et une mesure de similarité	aucune	Dimension de l'espace de projection	Taille des communautés
Complexité	$O(IKM)$	$O(M^2N)$	$O(N^3)$	$O(IN)$
Déterministe	non	oui	oui	oui
Identification de communautés de tailles et/ou de densités différentes	oui*	oui*	oui*	non
Multi-vue sur la structure de communautés	non	non	non	non

Tableau 3.1 – Propriétés des algorithmes : K-moyennes (KM), Girvan&Newman (GN), Donettei&Munoz (DM) et Bissection spectrale (BS)
(*I* : nombre d'itérations, *K* : nombre de communautés, *M* : nombre total de liens, *N* : nombre de sommets)

Propriété	**MNL**	**PHITS**	**SPAEM**	**SBM**
Identification de communautés dans des graphes orientés	oui	oui	non	oui
Identification de communautés qui se recouvrent	non*	oui	oui	non*
Connaissances a priori	Nombre de communautés	Nombre de communautés	Nombre de communautés*	Nombre de communautés*
Complexité	$O(IKM)$	$O(IKM)$	$O(IKM)$	$O(IK^2N^2)$
Déterministe	non	non	non	non
Identification de communautés de tailles et/ou de densités différentes	oui	oui	oui	oui
Multi-vue sur la structure de communautés	non	oui	non	non

Tableau 3.2 – Propriétés des modèles MNL, PHITS, SPAEM et SBM
(*I* : nombre d'itérations, *K* : nombre de communautés, *M* : nombre total de liens, *N* : nombre de sommets)

L'algorithme de bissection spectrale requiert en entrée la taille des communautés pour pouvoir les détecter. En pratique, cette information est généralement inconnue. Le calcul du premier vecteur propre non trivial de la matrice de Laplace est généralement réalisé avec la méthode de Lancoz. Celle-ci possède une complexité linéaire par rapport au nombre de nœuds dans le graphe, ce qui permet de l'utiliser avec des graphes de grande taille.

Pour SPAEM, Ren et al. [Ren et al. 09] ont proposé une méthode basée sur le critère MDL (Minimum Description Length) pour déterminer de manière automatique le nombre de communautés. Daudin et al. [Daudin et al. 08] proposent plutôt un SBM qui utilise le critère ICL (Integrated Classification Likelihood) pour trouver le nombre de communautés. Ces deux critères sont basés sur la vraisemblance des données qui, par définition, croit lorsque la complexité du modèle (i.e. le nombre de communautés) augmente. Ces critères tendent alors à favoriser des modèles complexes.

Les quatre modèles génératifs présentés ainsi que l'algorithme des K-moyennes sont des méthodes itératives qui partent d'une structure de communautés initiale qu'ils améliorent à chaque itération. La qualité de la structure de communautés finale dépend toutefois de la qualité de la structure de communautés initiale. D'autre part, les méthodes MNL, PHITS, SPAEM et KM (K-moyennes) ont une complexité linéaire par rapport au nombre de liens dans le graphe, à la différence du modèle SBM qui a une complexité quadratique par rapport au nombre de sommets. La forte complexité de ce dernier est due au fait qu'il modélise la génération de toutes relations entre sommets (i.e. celles de la présence et de l'absence de liens), tandis que les modèles MNL, PHITS et SPAEM ne modélisent que la présence de liens et tirent ainsi profit de la faible densité du graphe.

En considérant les critères 1, 3 et 4 (que nous jugeons les plus importants), nous remarquons que l'algorithme des K-moyennes et les deux modèles MNL et PHITS sont les mieux adaptés à l'identification de communautés dans les graphes de documents. Le modèle PHITS possède en plus l'avantage de prendre en compte le recouvrement des communautés. En fait, la plupart des algorithmes d'ISC existants sont destinés à l'identification de communautés disjointes dans des graphes non-orientés [Fortunato 10].

Finalement, nous montrons dans le chapitre 4 que les modèles MNL et PHITS sont sujets à un problème important qui se manifeste par de très mauvaises performances lorsque la densité du graphe à analyser est très faible, alors que l'algorithme des K-moyennes n'est pas sensible à cette faible densité. Nous avons remarqué ce problème également avec les modèles SPAEM et SBM lors de l'analyse de graphes non-orientés de très faible densité. Il s'agit d'un problème qui concernerait apparemment toutes les approches génératives existantes.

Nous proposons dans le chapitre suivant deux nouveaux modèles génératifs pour l'identification de communautés dans les graphes de documents. Nous proposons également pour ces modèles des techniques pour pallier au problème de la faible densité, pour déterminer de manière automatique le nombre de communautés et enfin des techniques d'initialisation car cette étape est cruciale pour ces modèles.

4

Des Modèles Génératifs pour l'Identification de Structures de Communautés dans les Graphes de Documents

Dans cette thèse, nous défendons l'idée que les modèles génératifs représentent une solution très intéressante au problème de l'identification de structures de communautés. Nous avons montré à la fin du chapitre précédent que ce type de modèles possède plusieurs avantages tels que la capacité à analyser des graphes orientés ou encore la détection de communautés qui se recouvrent. Cependant, en utilisant ces modèles pour l'ISC dans des graphes de documents, nous avons trouvé que ces modèles donnent de mauvais résultats par rapport à des méthodes classiques telles que l'algorithme des K-moyennes. Cette mauvaise performance est en fait due à la faible densité en liens qui caractérise les graphes de documents.

Dans ce chapitre, nous proposons le modèle SPCE pour l'identification de structures de communautés dans des graphes orientés. Contrairement aux autres approches génératives, SPCE est robuste à la faible densité des graphes de documents. Cette robustesse est obtenue par l'utilisation de la technique du lissage ainsi que d'une initialisation adéquate de l'algorithme d'estimation des paramètres du modèles SPCE. Dans le but de valider le modèle SPCE, nous l'avons comparé à d'autres approches génératives et non génératives en utilisant quatre graphes de documents. Les résultats expérimentaux montrent que le modèle SPCE réalise les meilleures performances par rapport à toutes les autres approches étudiées. A la fin du chapitre, nous proposons le modèle SPCE-PLSA pour l'identification de thématiques dans les collections de documents. Il s'agit d'une extension du modèle SPCE permettant de prendre en compte non seulement les liens entre documents mais aussi leurs contenus. Ce modèle est également évalué expérimentalement en utilisant deux corpus de documents. Les résultats obtenus montrent que le fait de combiner les liens et les contenus permet d'améliorer

considérablement l'identification de thématiques par rapport à une approche basée uniquement sur les contenus des documents.

4.1 Le modèle SPCE

Nous présentons le modèle SPCE, un nouveau modèle génératif pour l'identification de structures de communautés. La motivation initiale de ce modèle réside dans les faibles performances des modèles génératifs existants pour l'ISC lorsqu'on les a appliqués à des graphes de documents. Nous avons remarqué que ces modèles donnaient de très mauvaises performances lorsque le graphe à analyser avait une faible densité tandis que de bonnes performances étaient obtenues lorsque le graphe contient "beaucoup" ou "suffisamment" de liens. Il est bien connu, dans la littérature sur les modèles probabilistes, que ce type de modèles atteint ses limites avec des données creuses (i.e. contenant beaucoup de zéros) [Agresti 07]. L'estimation de paramètres à partir de telles données s'avère être de mauvaise qualité [Brown and Fuchs 83][Dhillon and Guan 03].

Par exemple, dans [Popescul et al. 01], les auteurs utilisent un modèle génératif inspiré du modèle PLSA de Hofmann pour une application de systèmes de recommandation. Ils rapportent qu'en raison de la "sparsité" des données qu'ils utilisent, leur modèle donne des résultats beaucoup moins bons que d'autres méthodes simples et non probabilistes. Ils ont alors proposé "d'enrichir" leur matrice de données en rajoutant des informations basées sur un calcul de similarité entre objets. En rendant la matrice plus dense, ils montrent que leur modèle donne de bien meilleurs résultats notamment en évitant les "mauvais" maximums locaux lors de l'estimation des paramètres.

Afin de pallier au problème de la faible densité des graphes de documents, nous proposons le modèle SPCE qui est basé sur la mise en œuvre du lissage et sur une bonne initialisation. Nous pensons en effet que ces deux points (i.e. le lissage et l'initialisation) permettent au modèle d'éviter les maximums locaux de mauvaise qualité. La technique du lissage est une méthode très utilisée dans le cadre de l'analyse de tableaux de contingence contenant beaucoup de zéros [Simonoff 98][Dahinden et al. 07]. Elle est également utilisée en recherche d'information [Manning et al. 08] ou dans les modèles probabilistes de la langue [Rigouste 06].

4.1.1 Processus génératif

Le processus génératif du modèle SPCE [Chikhi et al. 09][Chikhi et al. 10] est proche de celui de PHITS (et donc de celui de PLSA) à ceci près qu'il utilise des priors (i.e. des distributions a priori) sur les paramètres du modèle. Le but de ces priors est de lisser les distributions des paramètres.

Soit $G = (V, E)$ un graphe orienté d'ordre N représenté par sa matrice d'adjacence **A**. Le modèle SPCE suppose que les M liens de ce graphe sont générés par K communautés en utilisant le processus suivant :

i. Pour $i = 1$ à N faire :

1. Tirer un vecteur de paramètres $\mu_i \sim \mathrm{Dir}(\alpha)$ où α est un hyper-paramètre de la loi de Dirichlet et $\sum_{k=1}^{K} \mu_{ki} = 1$.

ii. Pour $k = 1$ à K faire :

1. Tirer un vecteur de paramètres $\phi_k \sim \mathrm{Dir}(\beta)$ où β est un hyper-paramètre de la loi de Dirichlet et $\sum_{i=1}^{N} \phi_{ik} = 1$.

iii. Pour $m = 1$ à M faire :

1. Choisir un nœud source $s_m \sim \mathrm{Mult}(1, (\pi_1, \ldots, \pi_N))$ où π est un vecteur de paramètres tel que $\sum_{i=1}^{N} \pi_i = 1$.
2. Conditionnellement à s_m, choisir une communauté $c_m \sim \mathrm{Mult}\left(1, (\mu_{1s_m}, \ldots, \mu_{Ks_m})\right)$.
3. Conditionnellement à c_m, choisir un nœud destination $d_m \sim \mathrm{Mult}\left(1, (\phi_{1c_m}, \ldots, \phi_{Nc_m})\right)$.
4. Générer un lien entre le nœud s_m et le nœud d_m

Nous noterons par Θ l'ensemble des paramètres (et hyper-paramètres) du modèle SPCE i.e. :

$$\Theta = \left\{ (\pi_i)_{i=1,\ldots,N}, (\mu_{ki})_{\substack{k=1,\ldots,K \\ i=1,\ldots,N}}, (\phi_{jk})_{\substack{j=1,\ldots,N \\ k=1,\ldots,K}}, \alpha, \beta \right\}$$

La figure 4.1 indique la représentation graphique du modèle SPCE. Ce modèle est basé sur des hypothèses similaires à celles de PHITS i.e. :

i) La distribution jointe d'un lien $\{(s_m, d_m)\}$ et de sa communauté c_m est :

$$\begin{aligned} p(S = s_m, D = d_m, C = c_m \,;\Theta) &= p(S = s_m \,;\Theta)\, p(C = c_m \mid S = s_m \,;\Theta) \\ &\quad p(D = d_m \mid C = c_m \,;\Theta) \\ &= \pi_{s_m} \mu_{c_m s_m} \phi_{d_m c_m} \end{aligned} \tag{4.1}$$

ii) Les liens observés $\{(s_m, d_m)\}$ sont indépendants les uns des autres. La log-vraisemblance du graphe observé est donc :

$$\begin{aligned} LL &= \log p\left((s_1, d_1), \ldots, (s_M, d_M) \,;\Theta\right) \\ &= \sum_{m=1}^{M} \log p(S = s_m, D = d_m \,;\Theta) \\ &= \sum_{m=1}^{M} \log \sum_{k=1}^{K} p(S = s_m, D = d_m, C = k \,;\Theta) \\ &= \sum_{m=1}^{M} \log \sum_{k=1}^{K} \left(\pi_{s_m} \mu_{ks_m} \phi_{d_m k}\right) \end{aligned} \tag{4.2}$$

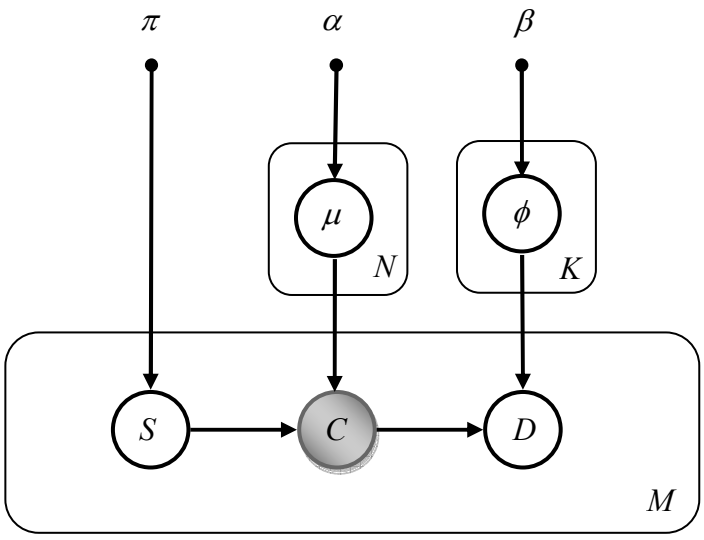


Figure 4.1 - Représentation graphique du modèle SPCE

Concernant les priors sur les paramètres, nous avons utilisé des distributions de Dirichlet car celles-ci sont conjuguées à la loi multinomiale. Cela permet en fait de simplifier la procédure d'estimation des paramètres du modèle SPCE. Plus précisément, en faisant un tel choix, la distribution a posteriori des paramètres aura la même forme que la distribution a priori. De plus, nous considérons que les distributions de Dirichlet utilisées sont symétriques i.e. les valeurs de leur vecteur de paramètres sont toutes égales ($\alpha_1 = \ldots = \alpha_K$ et $\beta_1 = \ldots = \beta_N$).

4.1.2 Estimation des paramètres

Pour l'estimation des paramètres du modèle SPCE, nous utilisons la méthode d'estimation par maximum a posteriori (MAP). Ainsi, au lieu de maximiser la vraisemblance des données (comme c'est le cas de PHITS), nous maximisons plutôt l'expression suivante qui représente la distribution a postériori des paramètres :

$$p\left(\mu,\phi \mid (s_1,d_1),\ldots,(s_M,d_M)\right) \propto p\left((s_1,d_1),\ldots,(s_M,d_M)\right) p(\mu \mid \alpha)\, p(\phi \mid \beta) \qquad (4.3)$$

En raison de la présence de variables cachées, l'expression (4.3) ne peut être maximisée analytiquement. Nous utilisons alors l'algorithme EM qui nous permet d'avoir un algorithme simple pour le calcul des paramètres du modèle SPCE.

La log-vraisemblance des données complètes (i.e. celle des triplets $\{(s_m,d_m,c_m)\}$) est :

$$\begin{aligned} LL^C &= \log\left(p\left((s_1,d_1,c_1),\ldots,(s_M,d_M,c_M)\ ;\Theta\right)\right) \\ &= \sum_{m=1}^{M} \log p\left(S=s_m, D=d_m, C=c_m\ ;\Theta\right) \\ &= \sum_{m=1}^{M}\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{K} \mathbf{1}_{\{s_m=i,d_m=j,c_m=k\}} \log p\left(S=i,D=j,C=k\ ;\Theta\right) \\ &= \sum_{m=1}^{M}\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{K} \mathbf{1}_{\{s_m=i,d_m=j,c_m=k\}} \left(\log\pi_i + \log\mu_{ki} + \log\phi_{jk}\right) \end{aligned} \qquad (4.4)$$

La distribution a posteriori des communautés est obtenue en appliquant la formule de Bayes. Pour $i=1\ldots N,\ j=1\ldots N,\ k=1\ldots K$, cette distribution est donnée par :

$$\begin{aligned}\omega_{kij} &= p\left(C=k \mid S=i, D=j\ ;\Theta^{old}\right)\\ &= \frac{p\left(S=i, D=j, C=k\ ;\Theta^{old}\right)}{p\left(S=i, D=j\ ;\Theta^{old}\right)}\\ &= \frac{\mu_{ki}^{old}\phi_{jk}^{old}}{\sum_{t=1}^{K}\mu_{ti}^{old}\phi_{jt}^{old}}\end{aligned} \tag{4.5}$$

L'espérance conditionnelle de la log-vraisemblance des données complètes est donc :

$$\begin{aligned}Q &= \mathbf{E}_C\left[\sum_{m=1}^{M}\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{K}\mathbf{1}_{\{s_m=i,d_m=j,c_m=k\}}\left(\log\pi_i+\log\mu_{ki}+\log\phi_{jk}\right)\right]\\ &= \sum_{m=1}^{M}\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{K}\mathbf{1}_{\{s_m=i,d_m=j\}}\mathbf{E}_{C_m}\left[\mathbf{1}_{\{c_m=k\}}\right]\left(\log\pi_i+\log\mu_{ki}+\log\phi_{jk}\right)\\ &= \sum_{m=1}^{M}\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{K}\mathbf{1}_{\{s_m=i,d_m=j\}}p\left(C=k \mid S=i, D=j\ ;\Theta^{old}\right)\left(\log\pi_i+\log\mu_{ki}+\log\phi_{jk}\right)\\ &= \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{K}A_{ij}\omega_{kij}\left(\log\pi_i+\log\mu_{ki}+\log\phi_{jk}\right)\\ &= \sum_{i=1}^{N}\sum_{j=1}^{N}A_{ij}\log\pi_i+\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{K}A_{ij}\omega_{kij}\left(\log\mu_{ki}+\log\phi_{jk}\right)\end{aligned} \tag{4.6}$$

L'estimation par maximum a posteriori avec l'algorithme EM diffère de l'estimation par maximum de vraisemblance du fait que ce n'est pas la quantité Q qui est maximisée à l'étape M, mais plutôt la quantité Q^{MAP} suivante [Bishop 07] :

$$Q^{MAP} = Q + \log p(\theta) \tag{4.7}$$

où $p(\theta)$ est la distribution à priori de paramètres θ du modèle.

Pour SPCE, la quantité Q^{MAP} est donnée par :

$$\begin{aligned}Q^{MAP} &= Q+\log p\left(\mu,\phi \mid \alpha,\beta\right)\\ &= Q+\log p\left(\mu \mid \alpha\right)+\log p\left(\phi \mid \beta\right)\\ &= Q+\sum_{i=1}^{N}\log p\left(\mu_i \mid \alpha\right)+\sum_{k=1}^{K}\log p\left(\phi_k \mid \beta\right)\end{aligned} \tag{4.8}$$

Les variables $\{\mu_i\}$ et $\{\phi_k\}$ étant des variables qui suivent une loi de Dirichlet, leurs distributions sont données respectivement par [Bishop 07] :

$$p\left(\mu_i \mid \alpha\right) = C(\alpha)\prod_{k=1}^{K}\mu_{ki}^{\alpha-1} \tag{4.9}$$

$$p(\phi_k \mid \beta) = C(\beta) \prod_{j=1}^{N} \phi_{jk}^{\beta-1} \tag{4.10}$$

où $C(\alpha)$ (resp. $C(\beta)$) est un constante dont la valeur ne dépend que de α (resp. β).

En remplaçant (4.9) et (4.10) dans (4.8) nous obtenons :

$$\begin{aligned} Q^{MAP} &= Q + \sum_{i=1}^{N}\sum_{k=1}^{K}(\alpha-1)\log\mu_{ki} + \sum_{k=1}^{K}\sum_{j=1}^{N}(\beta-1)\log\phi_{jk} + const \\ &= \sum_{i=1}^{N}\sum_{j=1}^{N} A_{ij}\log\pi_i + \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{K} A_{ij}\omega_{kij}\left(\log\mu_{ki} + \log\phi_{jk}\right) \\ &+ \sum_{i=1}^{N}\sum_{k=1}^{K}(\alpha-1)\log\mu_{ki} + \sum_{k=1}^{K}\sum_{j=1}^{N}(\beta-1)\log\phi_{jk} + const \end{aligned} \tag{4.11}$$

Les nouveaux paramètres Θ qui maximisent Q^{MAP} sont obtenus en maximisant le lagrangien suivant :

$$\begin{aligned} H = &\sum_{i=1}^{N}\sum_{j=1}^{N} A_{ij}\log\pi_i + \sum_{i=1}^{N}\sum_{j=1}^{N} A_{ij}\sum_{k=1}^{K}\omega_{kij}\left(\log\mu_{ki} + \log\phi_{jk}\right) + \sum_{i=1}^{N}\sum_{k=1}^{K}(\alpha-1)\log\mu_{ki} \\ &+ \sum_{k=1}^{K}\sum_{j=1}^{N}(\beta-1)\log\phi_{jk} + \lambda\left(1-\sum_{i=1}^{N}\pi_i\right) + \sum_{i=1}^{N}\sigma_i\left(1-\sum_{k=1}^{K}\mu_{ki}\right) + \sum_{k=1}^{K}\zeta_k\left(1-\sum_{j=1}^{N}\phi_{jk}\right) \end{aligned} \tag{4.12}$$

où λ, $(\sigma_1,\ldots,\sigma_N)$ et $(\zeta_1,\ldots,\zeta_K)$ sont les multiplicateurs de Lagrange.

En résolvant l'équation $\frac{\partial H}{\partial \pi_i} = 0$, on obtient l'équation de ré-estimation suivante:

$$\pi_i = \frac{1}{N}\sum_{j=1}^{N} A_{ij} \tag{4.13}$$

Cela signifie que la probabilité a priori d'un nœud *i* est proportionnelle au nombre de liens sortants que possède ce nœud.

L'équation de ré-estimation de μ_{ki} est obtenue en résolvant l'équation $\frac{\partial H}{\partial \mu_{ki}} = 0$. On obtient ainsi :

$$\mu_{ki} = \frac{\alpha - 1 + \sum_{j=1}^{N} A_{ij}\omega_{kij}}{K(\alpha-1) + \sum_{t=1}^{K}\sum_{j=1}^{N} A_{ij}\omega_{tij}} \tag{4.14}$$

De même, la résolution de l'équation $\frac{\partial H}{\partial \phi_{jk}} = 0$ donne :

$$\phi_{jk} = \frac{\beta - 1 + \sum_{i=1}^{N} A_{ij}\omega_{kij}}{N(\beta-1) + \sum_{l=1}^{N}\sum_{i=1}^{N} A_{il}\omega_{kil}} \tag{4.15}$$

L'algorithme EM complet pour l'estimation des paramètres du modèle SPCE est décrit par l'algorithme 4.1. Nous constatons que dans l'étape M, α et β jouent le rôle de pseudo-comptes (ou de pseudo-liens) qui permettent à SPCE de prendre en compte la faible densité du graphe. Ainsi, lorsque $\alpha = \beta = 1$, SPCE est équivalent à PHITS. Ceci est évident car dans un pareil cas, les a priori de Dirichlet sont uniformes et les distributions a posteriori ne dépendent donc que de la vraisemblance. Afin d'éviter des valeurs de probabilité incohérentes, nous imposons $\alpha \geq 1$ et $\beta \geq 1$. En outre, dans le cas où $\alpha = \beta = 2$, cela revient au lissage de Laplace (ajout de 1) utilisé en recherche d'information.

Algorithme 4.1 : Algorithme d'estimation des paramètres du modèle SPCE

Entrée : - un graphe G d'ordre N représenté par sa matrice d'adjacence $\mathbf{A}$
- le nombre de communautés K
- les hyper-paramètres α et β

Sortie : les paramètres du modèle à savoir le vecteur $\boldsymbol{\pi}$ et les matrices $\boldsymbol{\mu}$ et $\boldsymbol{\varphi}$

début

// Initialisation

1. $\boldsymbol{\mu} \leftarrow \frac{\mathbf{1}_{K\times N}}{K}$, $\boldsymbol{\varphi} \leftarrow \frac{\mathbf{1}_{N\times K}}{N}$

// Optimisation

2. **pour** $i = 1 \ldots N$ **faire**
3. $\quad \pi_i = \frac{1}{N}\sum_{j=1}^{N} A_{ij}$
4. **fin**
5. **répéter**

// Etape E de l'algorithme

6. **pour** $i = 1 \ldots N$, $j = 1 \ldots N$ et $k = 1 \ldots K$ **faire**
7. $\quad \omega_{kij} \leftarrow \frac{\mu_{ki}\phi_{jk}}{\sum_{t=1}^{K}\mu_{ti}\phi_{jt}}$
8. **fin**

// Etape M de l'algorithme

9. **pour** $i = 1 \ldots N$ et $k = 1 \ldots K$ **faire**
10. $\quad \mu_{ki} = \frac{\alpha - 1 + \sum_{j=1}^{N} A_{ij}\omega_{kij}}{K(\alpha - 1) + \sum_{t=1}^{K}\sum_{j=1}^{N} A_{ij}\omega_{tij}}$
11. $\quad \phi_{jk} = \frac{\beta - 1 + \sum_{i=1}^{N} A_{ij}\omega_{kij}}{N(\beta - 1) + \sum_{l=1}^{N}\sum_{i=1}^{N} A_{il}\omega_{kil}}$
12. **fin**
13. **jusqu'à** *convergence*

fin

4.2 Mise en œuvre du modèle SPCE

4.2.1 Initialisation de l'algorithme EM

Comme nous l'avons précisé, l'initialisation joue un rôle important dans l'algorithme EM et dans les techniques d'optimisation en général. C'est pourquoi nous proposons ici trois stratégies d'initialisation différentes pour l'algorithme d'estimation des paramètres du modèle SPCE. Nous avons tenu à ce que ces stratégies aient une complexité faible. En effet, il n'est guère intéressant d'avoir une méthode d'initialisation dont la complexité est supérieure à celle de l'algorithme SPCE.

La première méthode d'initialisation est une initialisation aléatoire où les paramètres $\boldsymbol{\mu}$ et $\boldsymbol{\varphi}$ sont tirés à partir d'une distribution de Dirichlet. Nous avons pour cela utilisé la boîte à outils Fastfit développée par Thomas Minka [URL 2]. Nous désignerons par $SPCE_R$ la version de SPCE qui utilise cette méthode d'initialisation.

La deuxième méthode consiste à utiliser l'algorithme des *K*-moyennes pour faire un premier regroupement des documents. Ce regroupement est ensuite utilisé pour initialiser la matrice $\boldsymbol{\varphi}$. Plus précisément, il s'agit d'initialiser les *K* colonnes de cette matrice par les *K* centres (ou centroïdes) des clusters. Pour le calcul de similarités entre les documents par l'algorithme des K-moyennes, nous utilisons la mesure du cosinus. Nous désignerons par $SPCE_K$ la version de SPCE qui utilise une initialisation basée sur l'algorithme des K-moyennes.

La troisième stratégie est basée sur deux éléments. D'une part, l'utilisation de Graclus [Dhillon et al. 07], un algorithme efficace (en termes de qualité et de rapidité) pour le partitionnement de graphes, et d'autre part, l'utilisation de la mesure d'Amsler [Amsler 72] pour le calcul de similarité entre documents. Graclus est un algorithme clustering spectral de graphes. Cependant, comme il ne fonctionne qu'avec des graphes non-orientés, nous calculons à partir du graphe de documents initial un graphe non-orienté en utilisant la mesure d'Amsler. La mesure d'Amsler entre deux documents *i* et *j* est définie par [Calado et al. 03] :

$$sim^{Ams}(i,j) = \frac{\left|V(i) \cap V(j)\right|}{\left|V(i) \cup V(j)\right|} \tag{4.16}$$

où *V(k)* correspond au voisinage immédiat du document *i* i.e. l'ensemble des documents qui pointent vers *i* ou bien qui sont pointés par celui-ci. L'avantage de cette mesure de similarité est qu'elle prend en compte à la fois les liens entrants et les liens sortants pour calculer la similarité entre deux documents, à la différence des mesures de co-citation ou de couplage bibliographique qui ne prennent en compte qu'un seul type de liens (sortants ou entrants). De la même façon que la méthode précédente, une fois le partitionnement effectué par Graclus, nous initialisions la matrice $\boldsymbol{\varphi}$ par les centres des différentes partitions. Nous désignerons par $SPCE_G$ la version de SPCE qui utilise cette troisième méthode d'initialisation.

4.2.2 Estimation des paramètres de lissage

Nous avons remarqué lors de nos expérimentations que les paramètres de lissage α et β influent beaucoup sur la qualité de la structure de communautés identifiée par SPCE. C'est pourquoi nous avons jugé utile de proposer une méthode permettant de déterminer les valeurs optimales pour ces paramètres. Dans la version initiale de SPCE, publiée à ICMLA'09, nous avons utilisé la modularité pour trouver les valeurs des paramètres de lissage. Cependant, nous allons ici proposer une nouvelle méthode pour arriver à cette fin et laisser plutôt la modularité comme mesure d'évaluation qui sera utilisée dans la section 4.3.

Nous commençons d'abord par faire une simplification en considérant que $\alpha = \beta$. Cette simplification est motivée par le fait que l'hyper-paramètre α (resp. β) détermine le nombre total de pseudo-liens sortants (resp. entrants) qui seront pris en compte lors de l'estimation des paramètres du modèle. En prenant $\alpha = \beta$, nous supposons ainsi que le nombre de pseudo-liens entrants est égal au nombre de pseudo-liens sortants. Cela semble raisonnable puisque dans un graphe orienté, le nombre total de liens entrants est toujours égal au nombre total de liens sortants. De plus, nous allons supposer que α et β prennent des valeurs telles que :

$$\alpha = \beta = 1 + \lambda \frac{M}{K \times N} \tag{4.17}$$

où *M* est le nombre total de liens, *K* est le nombre de communautés, *N* est le nombre total de nœuds (i.e. de documents) et λ est un réel compris entre 0 et 1. Le paramètre λ détermine le pourcentage de pseudo-liens par rapport au nombre total de liens qui seront pris en compte lors de l'estimation des paramètres. $\lambda = 0$ signifie qu'aucun pseudo-lien ne sera pris en compte (i.e. pas de lissage) tandis que $\lambda = 1$ signifie que *M* pseudo-liens (i.e. 100% du nombre total de liens) seront pris en compte.

La solution que nous proposons est basée sur la méthode de la validation croisée [Bishop 07] ainsi que sur la mesure de la perplexité comme critère de qualité d'un modèle. La validation croisée à *K*-passes (*K*-fold cross validation) est une technique permettant de sélectionner le meilleur modèle parmi plusieurs modèles candidats (ces modèles ne sont pas forcément des modèles probabilistes) [Utsugi 97]. Elle consiste à diviser l'ensemble de données en *K* ensembles de tailles plus ou moins égales puis à répéter *K* fois l'opération suivante : utiliser *K*-1 groupes (appelés données d'apprentissage) pour l'apprentissage du modèle et le groupe restant (appelé donnée de test) pour l'évaluation du modèle avec un critère donné. La qualité du modèle est alors égale à la moyenne des *K* mesures de qualités calculées lors de chaque passe.

La perplexité est une mesure propre aux modèles génératifs. Sa valeur est inversement proportionnelle à la vraisemblance des données de test. Formellement, la perplexité d'un ensemble de données de test *T* est définie par [Hofmann 01]:

$$PP = \exp\left(-\frac{LL(T)}{|T|} \right) \tag{4.18}$$

où *LL(T)* correspond à la log-vraisemblance de l'ensemble de test *T*, et $|T|$ correspond à la taille de cet ensemble.

Pour le modèle SPCE, la perplexité d'un ensemble de test $T = \{(s_1, d_1), \ldots, (s_H, d_H)\}$ où H correspond au nombre d'éléments (i.e. de liens) de cet ensemble est définie par :

$$PP^{SPCE} = \exp\left(-\frac{\log p\left((s_1, d_1), \ldots, (s_H, d_H) \mid \Theta\right)}{H}\right) = \exp\left(-\frac{\sum_{h=1}^{H} \log \sum_{k=1}^{K} \left(\pi_{s_h} \mu_{k s_h} \phi_{d_h k}\right)}{H}\right) \quad (4.19)$$

L'interprétation de la perplexité est la suivante : plus elle est petite, meilleur est le modèle.

4.2.3 Calcul du nombre de communautés

Dans la littérature sur les modèles génératifs, il existe tout un domaine qui s'intéresse à la problématique de la sélection de modèles. Plusieurs critères ont été proposés tels que les critères AIC (Akaike Information Criterion) ou BIC (Bayesian Information Criterion) [MacKay 02]. L'avantage de ces mesures est qu'elles peuvent être calculées directement à partir de la log-vraisemblance des données. Cependant, cet avantage s'avère être aussi un inconvénient car la vraisemblance des données d'apprentissage n'est pas un critère suffisant pour juger de la qualité d'un modèle génératif. D'ailleurs, lors de nos expérimentations, nous avons testé la capacité du modèle SPCE à déterminer le nombre de communautés en utilisant les critères AIC et BIC mais nous n'avons pas obtenus de résultats concluants.

Afin de calculer le nombre de communautés avec le modèle SPCE, nous adoptons à nouveau la méthode de la validation croisée décrite dans la section précédente. Nous appliquons ainsi l'algorithme SPCE en utilisant différentes valeurs pour le nombre de communautés. Pour chacune de ces valeurs, la perplexité du modèle obtenu est calculée et le modèle aboutissant à la plus faible perplexité est choisi.

4.3 Evaluation expérimentale du modèle SPCE

4.3.1 Evaluation de l'ISC

a) Graphes utilisés :

Notre première tâche lors de l'évaluation va concerner l'identification de structures de communautés dans des graphes de documents. Nous utilisons pour cela quatre graphes de documents : Cora, Citeseer, Plsam_Physics et Solar_Wind (voir les caractéristiques de ces graphes à la section 2.5.1).

Nous utilisons à chaque fois le graphe initial et le graphe transposé pour l'évaluation. Les graphes de documents étant orientés, le regroupement des documents en communautés peut se faire soit en utilisant les liens entrants soit en utilisant les liens sortants. Nous considérerons à chaque fois les deux cas. Rappelons que les modèles PHITS et SPCE permettent d'avoir un

regroupement à partir des liens entrants et à partir des liens sortants. Pour les autres approches, il sera nécessaire de lancer l'algorithme deux fois : une première fois avec la matrice d'adjacence et une deuxième fois avec la transposée de la matrice d'adjacence.

Pour chacun des graphes, nous analysons uniquement la plus grande composante faiblement connexe.

b) Mesures d'évaluation :

Pour les mesures d'évaluation, plusieurs critères sont envisageables. En effet, l'ISC peut être considérée comme étant une tâche de clustering et dans ce domaine, les chercheurs ont proposé depuis longtemps plusieurs mesures d'évaluation de clustering. Ces mesures d'évaluation sont de deux types : les mesures internes et les mesures externes. Le lecteur est invité à consulter [Jain et al. 99][Halkidi et al. 01][Tan et al. 05] pour plus de détail sur ces mesures.

Les mesures internes sont propres au modèle, nous pouvons citer la mesure de silhouette, la distance euclidienne ou encore la vraisemblance des données. D'autres mesures internes concernent l'identification de communautés telles que la modularité de Newman ou encore la coupe normalisée (normalized cut [Shi and Malik 97]). Parmi toutes ces mesures, nous utiliserons la modularité qui est d'ailleurs la plus utilisée [Fortunato 10].

Les mesures dites externes font appel à des informations supplémentaires concernant la classe de chaque objet à classer. Dans le cas de l'ISC, il s'agit d'informations concernant la communauté à laquelle appartient chaque sommet. Là encore, une panoplie de mesures existe, nous citerons par exemple : l'entropie, la pureté, la F-mesure, l'information mutuelle normalisée (NMI, Normalized Mutual Information), l'indice de Rand, la variation d'information. etc. Nous avons utilisé plusieurs de ces mesures lors de nos expérimentations mais nous présenterons uniquement les résultats concernant la F-mesure et la NMI. Ces deux mesures sont en effet très utilisées dans le domaine de l'évaluation de la classification non supervisée.

La NMI est une mesure issue de la théorie de l'information permettant de comparer deux partitions (ou clusterings). Avec cette mesure, on considère chaque partition comme une distribution de probabilité et on calcule la quantité d'information commune aux deux distributions. Strehl [Strehl 02] propose de normaliser ce critère afin que sa valeur soit comprise entre 0 et 1. Une valeur 1 indique que les deux partitions sont identiques.

La F-mesure est une mesure très connue dans le domaine de la recherche d'information. Elle est égale à la moyenne géométrique entre la précision et le rappel. Elle est toujours comprise entre 0 et 1.

c) Algorithmes comparés :

Nous comparons plusieurs algorithmes d'ISC. L'accent sera mis sur les approches génératives. Mais pour donner une idée sur les performances des approches non génératives, nous présentons également les résultats de deux approches non génératives à savoir l'algorithme des K-moyennes et l'algorithme Graclus décrit dans la section 4.2.1. Pour les

approches génératives, nous comparons le modèle SPCE avec trois initialisations différentes au modèle PHITS et au modèle MNL (Mélange de Multinomiales de Newman et Leicht). Nous n'avons pas pu appliquer le modèle SBM avec nos graphes en raison de la forte complexité de ce modèle qui, rappelons-le, est de $O\left(IK^2N^2\right)$.

- *KM* : Algorithme des K-moyennes en utilisant la similarité de co-citation (resp. de couplage bibliographique) pour l'ISC en se basant sur les liens entrants (resp. sortants) des documents.

- *GR* : Algorithme Graclus de clustering spectral de graphes non orientés. Pour l'ISC en se basant sur les liens entrants (resp. sortants), un graphe de similarité entre documents est construit en utilisant la similarité de co-citation (resp. de couplage bibliographique) et donné en entrée à l'algorithme Graclus.

- *MNL* : Modèle de mélange de Multinomiales de Newman et Leicht. Les paramètres du modèle sont initialisés par des valeurs aléatoires.

- *PHITS* : Modèle Probabilistic HITS de Cohn et Hofmann. L'initialisation des paramètres se fait également par des valeurs aléatoires.

- $SPCE_R$, $SPCE_K$ et $SPCE_G$: il s'agit respectivement du modèle SPCE avec une initialisation aléatoire, basée sur K-means et basée sur Graclus pour les paramètres. Les valeurs optimales pour les paramètres de lissage sont calculées en utilisant la validation croisée comme indiqué dans la section 4.2.2. Nous présenterons les résultats obtenus pour les valeurs optimales des paramètres.

d) Résultats expérimentaux

Les tableaux 4.1, 4.2 et 4.3 indiquent les résultats expérimentaux obtenus en appliquant les différents algorithmes comparés avec les quatre graphes utilisés dans cette étude. Nous remarquons que les performances du modèle $SPCE_G$ sont meilleures que celles des autres approches ; elles sont en particulier nettement supérieures à celles des autres approches génératives. Les modèles MNL et PHITS donnent quant à eux de mauvais résultats. Les trois tableaux montrent également que l'initialisation du modèle SPCE influe beaucoup sur ses performances, et que l'initialisation basée sur l'algorithme Graclus et la mesure d'Amsler permet d'atteindre les meilleures performances. L'apport de l'initialisation est toutefois moins conséquent lors de l'analyse des graphes Plasma_Physics et Solar_Wind. En effet, ces deux graphes étant "suffisamment" denses, les différents modèles génératifs réussissent à trouver une structure de communautés de bonne qualité. Les résultats de PHITS et de $SPCE_R$ montrent que le lissage permet d'améliorer légèrement les performances et que celui-ci n'est pas suffisant à lui seul pour obtenir des performances optimales. Nous observons par ailleurs que les performances de l'algorithme Graclus sont comparables à celles de $SPCE_G$ (à une exception près pour le graphe Cora(T)). Enfin, nous noterons que les résultats avec les liens sortants sont dans la plupart des cas meilleurs que ceux avec les liens entrants. Cela s'explique par le fait que dans les quatre graphes étudiés, il existe plus de documents ayant au moins un lien sortant que de documents ayant au moins un lien entrant.

Tableau 4.1 - Résultats avec la NMI (Normalized Mutual Information) (O : Original, T : Transposé)

Graphe	KM	GR	MNL	PHITS	$SPCE_R$	$SPCE_K$	$SPCE_G$
Cora (O)	0.24	0.34	0.05	0.04	0.12	0.25	0.48
Cora (T)	0.27	0.05	0.10	0.07	0.16	0.30	0.51
Citeseer (O)	0.17	0.22	0.03	0.03	0.06	0.18	0.36
Citeseer (T)	0.20	0.28	0.08	0.04	0.07	0.23	0.42

Tableau 4.2 - Résultats avec la F-mesure (O : Original, T : Transposé)

Graphe	KM	GR	MNL	PHITS	$SPCE_R$	$SPCE_K$	$SPCE_G$
Cora (O)	0.42	0.57	0.25	0.24	0.32	0.43	0.64
Cora (T)	0.43	0.29	0.33	0.29	0.37	0.48	0.66
Citeseer (O)	0.39	0.48	0.30	0.29	0.33	0.40	0.62
Citeseer (T)	0.41	0.56	0.32	0.31	0.36	0.43	0.67

Tableau 4.3 - Résultats avec la modularité (O : Original, T : Transposé)

Graphe	KM	GR	MNL	PHITS	$SPCE_R$	$SPCE_K$	$SPCE_G$
Cora (O)	0.17	0.28	0.15	0.13	0.21	0.23	0.32
Cora (T)	0.28	0.03	0.14	0.21	0.30	0.34	0.46
Citeseer (O)	0.12	0.24	0.14	0.11	0.18	0.21	0.31
Citeseer (T)	0.19	0.29	0.10	0.15	0.23	0.26	0.39
Plasma_Physics (O)	0.34	0.40	0.22	0.28	0.34	0.38	0.39
Plasma_Physics (T)	0.46	0.41	0.37	0.38	0.48	0.52	0.54
Solar_Wind (O)	0.37	0.35	0.25	0.34	0.40	0.41	0.41
Solar_Wind (T)	0.47	0.47	0.37	0.44	0.49	0.52	0.52

4.3.2 Effet du paramètre de lissage

Nous étudions à présent l'effet du paramètre de lissage λ (lambda) sur la qualité des résultats du modèle SPCE. Nous reportons sur les figures 4.2, 4.3, 4.4 et 4.5 la modularité et la NMI en fonction du paramètre λ pour les graphes Cora et Citeseer. Les résultats avec les liens sortants étant similaires à ceux avec les liens entrants, nous reporterons ici uniquement les résultats concernant les liens entrants ; ceux avec les liens sortants sont présentés dans l'annexe C. L'évaluation de la perplexité montre que le lissage joue un rôle crucial lorsque l'initialisation n'est pas de bonne qualité alors qu'une bonne initialisation réduit l'importance du lissage. Les résultats de la NMI montrent que le lissage améliore légèrement les performances du modèle $SPCE_R$ tandis qu'à partir d'un certain seuil, il détériore légèrement celles de $SPCE_G$. De plus, pour ce dernier, nous avons remarqué qu'une valeur de $\lambda = 0.1$

donne toujours des résultats très satisfaisants. Cela nous permet de suggérer une valeur fixe pour λ avec le modèle $SPCE_G$ afin d'éviter le calcul de la valeur optimale de ce paramètre. Les figures montrent également qu'avec le modèle $SPCE_G$, la plus faible perplexité coïncide avec la meilleure modularité ainsi qu'avec la meilleure NMI. Cela confirme que la perplexité est aussi un bon indicateur pour la qualité de la structure de communautés identifiée.

Le tableau 4.4 indique les résultats de la NMI et de la perplexité obtenus en appliquant les algorithmes PHITS, $SPCE_R$, $SPCE_K$ et $SPCE_G$ avec les graphes Cora et Citeseer sans utiliser de lissage (i.e. $\lambda = 0$). Nous remarquons que les performances de PHITS sont les plus faibles alors que celles de $SPCE_G$ sont les meilleures. D'après la figure 4.3, la plus faible perplexité de $SPCE_G$ (lorsque $\lambda = 0.2$) avec le graphe Cora est égale à 4.8×10^7 tandis que lorsqu'aucun lissage n'est appliqué, ce même algorithme a une perplexité de 1.3×10^{10} (voir tableau 4.4). Cela signifie que le lissage a un impact très important sur la capacité du modèle SPCE à prédire de nouvelles données.

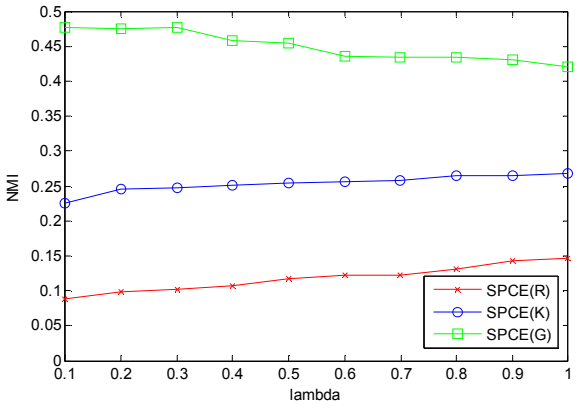


Figure 4.2 - La NMI en fonction du paramètre de lissage lambda avec le graphe Cora

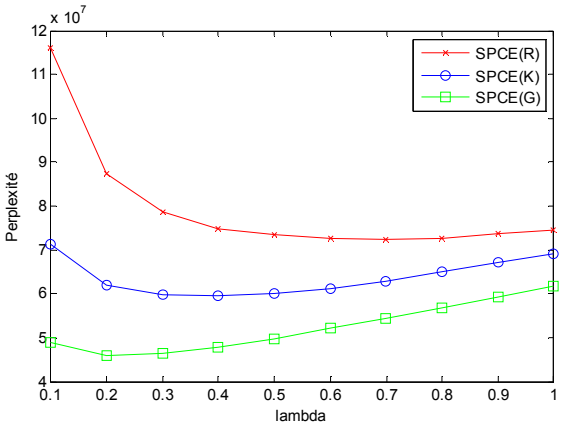


Figure 4.3 - La Perplexité en fonction du paramètre de lissage lambda avec le graphe Cora

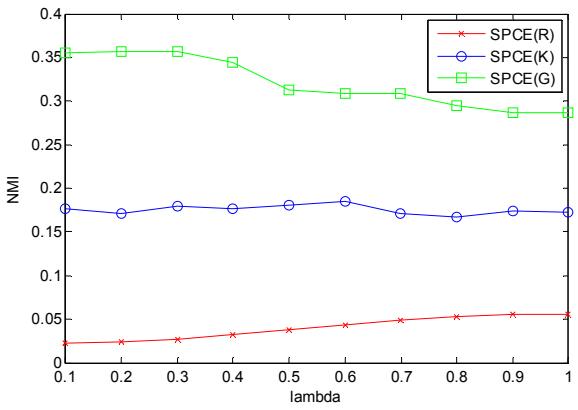


Figure 4.4 - La NMI en fonction du paramètre de lissage lambda avec le graphe Citeseer

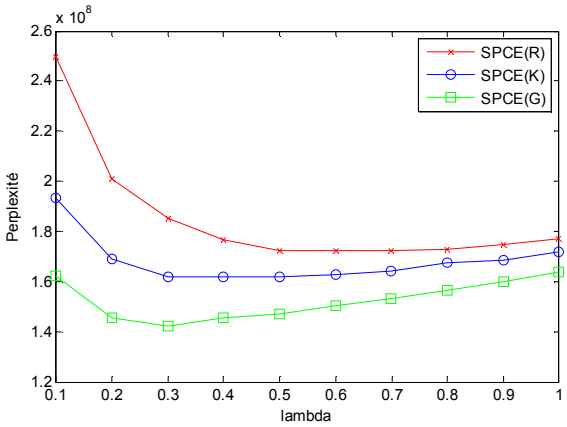


Figure 4.5 - La Perplexité en fonction du paramètre de lissage lambda avec le graphe Citeseer

Tableau 4.4 – Résultats de la NMI et de la perplexité avec $\lambda = 0$

	NMI				Perplexité			
Graphe	**PHITS**	**SPCE$_R$**	**SPCE$_K$**	**SPCE$_G$**	**PHITS**	**SPCE$_R$**	**SPCE$_K$**	**SPCE$_G$**
Cora (O)	0.04	0.04	0.20	0.46	4.4×10^{12}	4.4×10^{12}	1.2×10^{11}	1.3×10^{10}
Citeseer (O)	0.03	0.03	0.16	0.35	8.3×10^{11}	8.3×10^{11}	2.4×10^{10}	1.6×10^{10}

4.3.3 Evaluation de la robustesse à la faible densité

Nous évaluons ici la robustesse des modèles PHITS et SPCE face à la faible densité des graphes. Pour ce faire, nous analysons chacun des quatre graphes étudiés en retirant à chaque fois un certain pourcentage de liens. La modularité de la structure de communautés obtenue avec chacun des deux modèles est ensuite calculée. Les résultats obtenus sont indiqués par les figures 4.6, 4.7, 4.8 et 4.9. En retirant 40% des liens, la modularité de PHITS se dégrade de presque 50% tandis que celle de SPCE ne baisse que d'environ 10%. En enlevant 80% des liens, PHITS n'arrive plus à retrouver la structure de communautés puisque celle-ci a une modularité quasi nulle. La modularité avec SPCE quant à elle baisse d'environ 50% lorsque 80% des liens ont été retirés ce qui est satisfaisant. Notons également que le retrait de liens affecte beaucoup plus les graphes Cora et Citesser que les graphes Plasma_Physics et Solar_Wind : les deux premiers étant beaucoup moins denses que les deux derniers.

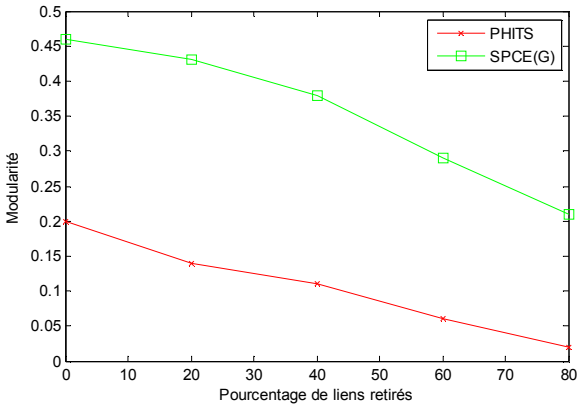


Figure 4.6 - La Modularité par rapport au pourcentage de liens retirés du graphe Cora

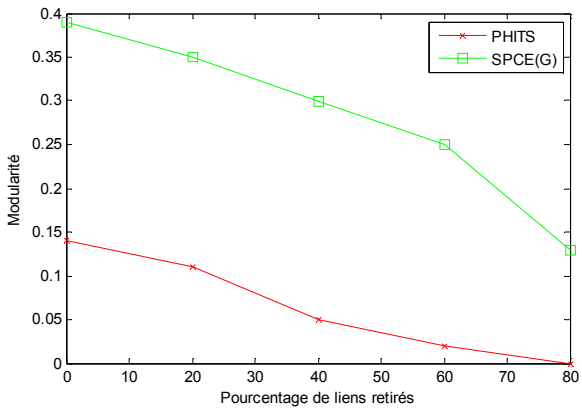


Figure 4.7 - La Modularité par rapport au pourcentage de liens retirés du graphe Citeseer

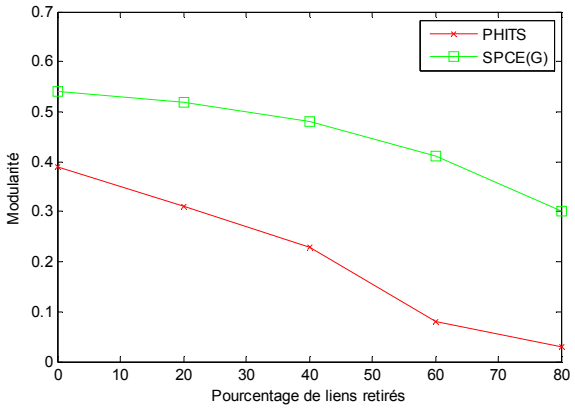


Figure 4.8 - La Modularité par rapport au pourcentage de liens retirés du graphe Plasma_Physics

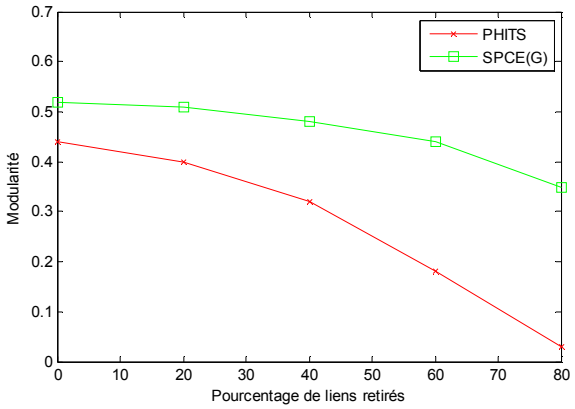


Figure 4.9 - La Modularité par rapport au pourcentage de liens retirés du graphe Solar_Wind

4.3.4 Evaluation de la convergence

Il est intéressant également d'étudier l'effet de l'initialisation sur la vitesse de convergence de l'algorithme d'estimation des paramètres du modèle SPCE. Nous avons pour cela étudier l'évolution de la valeur de la fonction objectif à maximiser par rapport au nombre d'itérations. Nous avons remarqué qu'une bonne initialisation permet à l'algorithme de converger plus vite i.e. en un nombre petit d'itérations. Un nombre d'itérations compris entre 30 et 50 itérations suffit en général à l'algorithme SPCE pour converger.

4.3.5 Evaluation du calcul du nombre de communautés

Nous avons suivi la méthodologie utilisée notamment par [Daudin et al. 08] et [Latouche et al. 10] pour évaluer la capacité de notre modèle à calculer le nombre correct de communautés dans un graphe. Cette méthode consiste à générer des graphes dont le nombre exact de communautés est connu. Cette méthode peut aussi être utilisée pour évaluer et comparer la qualité des algorithmes d'ISC ; les graphes générés servent de benchmarks. Lancichinetti et Fortunato [Lancichinetti and Fortunato 09] ont développé récemment un outil permettant de générer des graphes tests ayant différentes propriétés : orientés ou non, pondérés ou non, etc. Newman [Newman 04a][Newman 06] est le premier à avoir suggéré l'utilisation de graphes test pour comparer et évaluer des algorithmes d'ISC. Il a notamment proposé un processus permettant de générer un graphe non orienté de 128 nœuds appartenant à quatre communautés. Un paramètre permet de jouer sur la connectivité de la structure de communautés i.e. le nombre de liens que possède un nœud vers l'intérieur de la communauté versus vers l'extérieur de la communauté.

Pour l'évaluation de cette partie, nous avons généré des graphes orientés contenant un nombre fixe de communautés. Nous considérons un cas simple où les communautés sont de taille et de densité homogènes. Les résultats préliminaires obtenus montrent que la méthode proposée pour le calcul du nombre de communautés permet de déterminer le nombre exact de communautés dans un graphe.

Nous envisageons à l'avenir d'utiliser la plateforme de Lancichinetti et Fortunato [Lancichinetti and Fortunato 09] pour comparer et évaluer notre modèle avec différents types de graphes y compris en ce qui concerne l'évaluation du calcul du nombre de communautés. Il serait particulièrement intéressant d'étudier la sensibilité de SPCE au problème de la "résolution limite" dont souffre la modularité.

4.4 SPCE-PLSA : un modèle hybride pour l'analyse des liens et des contenus

Dans [Cohn and Hofmann 01], Hofmann, l'auteur du modèle PLSA et Cohn, l'un des auteurs du modèle PHITS, proposent le modèle PHITS-PLSA pour l'analyse combinée des liens et des contenus. En s'inspirant de leurs travaux, nous proposons dans cette section le modèle SPCE-PLSA qui combine notre modèle SPCE pour l'analyse de liens entre documents avec le modèle PLSA pour l'analyse des contenus des documents.

Les approches hybrides combinant analyse de liens et de contenus ont reçu une attention particulière de la part des chercheurs dans le domaine de la fouille de textes. A titre d'exemple, la thèse de Janssens [Janssens 07] est entièrement consacrée à cette problématique. D'autres exemples incluent les travaux de [Chakrabarti et al. 01], [Drost et al. 06], [Wang and Kitsuregawa 02], [Modha and Spangler 00] ou encore ceux de [Jo et al. 07]. Notons également à ce propos que nous avons proposé dans [Chikhi et al. 08b] une approche hybride pour le regroupement automatique de documents. Il s'agit dans ce cas d'une approche basée sur l'idée du voisinage bibliographique.

4.4.1 Processus génératif

Soit G un graphe de documents d'ordre N représenté par sa matrice d'adjacence **A**. Le modèle SPCE-PLSA suppose que les M liens de ce graphe et les H occurrences de mots dans les N documents sont générés par le processus suivant (V est la taille du vocabulaire et K est le nombre thèmes) :

i. Pour $i = 1$ à N faire :

1. Tirer un vecteur de paramètres $\mu_i \sim \mathrm{Dir}(\alpha)$ où α est un hyper-paramètre de la loi de Dirichlet et $\sum_{k=1}^{K} \mu_{ki} = 1$.

ii. Pour $k = 1$ à K faire :

1. Tirer un vecteur de paramètres $\phi_k^L \sim \mathrm{Dir}(\beta)$ où β est un hyper-paramètre de la loi de Dirichlet et $\sum_{i=1}^{N} \phi_{ik}^L = 1$.

iii. Pour $m = 1$ à M faire :

1. Choisir un nœud source $s_m \sim \text{Mult}\left(1,(\pi_1^L,\ldots,\pi_N^L)\right)$ où π^L est un vecteur de paramètres tel que $\sum_{i=1}^{N}\pi_i^L = 1$.
2. Conditionnellement à s_m, choisir un thème $t_m \sim \text{Mult}\left(1,(\mu_{1s_m},\ldots,\mu_{Ks_m})\right)$.
3. Conditionnellement à t_m, choisir un nœud destination $d_m \sim \text{Mult}\left(1,(\phi_{1t_m}^L,\ldots,\phi_{Nt_m}^L)\right)$.
4. Générer un lien entre le nœud s_m et le nœud d_m

iv. Pour $h = 1$ à H faire :

1. Choisir un document $x_h \sim \text{Mult}\left(1,(\pi_1^C,\ldots,\pi_N^C)\right)$ où π^C est un vecteur de paramètres tel que $\sum_{i=1}^{N}\pi_i^C = 1$.
2. Conditionnellement à x_h, choisir un thème $t_h \sim \text{Mult}\left(1,(\mu_{1x_h},\ldots,\mu_{Kx_h})\right)$.
3. Conditionnellement à t_h, choisir un mot $w_h \sim \text{Mult}\left(1,(\phi_{1t_h}^C,\ldots,\phi_{Vt_h}^C)\right)$ où ϕ^C est une matrice de paramètres de dimension $V \times K$ telle que $\forall k \in \{1,\ldots,K\}, \sum_{j=1}^{N}\phi_{jk}^C = 1$.
4. Rajouter le mot w_h au document x_h

Nous noterons par Θ l'ensemble des paramètres (et hyper-paramètres) du modèle SPCE-PLSA i.e. :

$$\Theta = \left\{\left(\pi_i^L\right)_{i=1,\ldots,N},\left(\pi_i^C\right)_{i=1,\ldots,N},\left(\mu_{ki}\right)_{\substack{k=1,\ldots,K\\i=1,\ldots,N}},\left(\phi_{jk}^L\right)_{\substack{j=1,\ldots,N\\k=1,\ldots,K}},\left(\phi_{vk}^C\right)_{\substack{v=1,\ldots,V\\k=1,\ldots,K}},\alpha,\beta\right\}$$

La figure 4.10 indique la représentation graphique du modèle SPCE-PLSA.

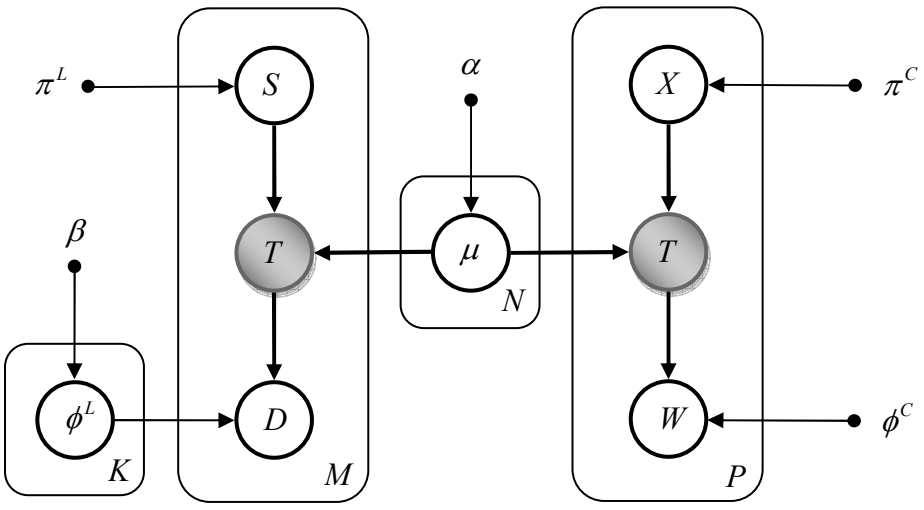


Figure 4.10 - Représentation graphique du modèle SPCE-PLSA

4.4.2 Estimation des paramètres

Comme pour le modèle SPCE, nous utilisons l'estimation par maximum a posteriori pour estimer les paramètres du modèle SPCE-PLSA. L'algorithme 4.2 indique l'algorithme EM pour l'estimation des paramètres de SPCE-PLSA. Il est à noter que ce dernier utilise un paramètre $0 \le \gamma \le 1$ indiquant l'importance relative donnée aux liens et aux contenus. Lorsque $\gamma = 0$, le modèle est équivalent à SPCE, tandis que lorsque $\gamma = 1$, le modèle revient à faire une analyse basée sur les contenus avec PLSA.

Algorithme 4.2 : Algorithme d'estimation des paramètres du modèle SPCE-PLSA

Entrée : - un graphe G d'ordre N représenté par sa matrice d'adjacence **A**
- une matrice documents-termes **W** de dimension $N \times V$
- le nombre de thématiques K
- les hyper-paramètres α et β
- le paramètre γ indiquant l'importance donnée aux liens et aux contenus

Sortie : les paramètres du modèle à savoir les vecteurs $\boldsymbol{\pi}^L$ et $\boldsymbol{\pi}^C$, et les matrices $\boldsymbol{\mu}$, $\boldsymbol{\varphi}^L$ et $\boldsymbol{\varphi}^C$

début

// Initialisation

1. $\boldsymbol{\mu} \leftarrow \frac{\mathbf{1}_{K\times N}}{K}$, $\boldsymbol{\varphi}^L \leftarrow \frac{\mathbf{1}_{N\times K}}{N}$, $\boldsymbol{\varphi}^C \leftarrow \frac{\mathbf{1}_{V\times K}}{V}$

// Optimisation

2. **pour** $i = 1 \ldots N$ **faire**
3. $\quad \pi_i^L = \frac{1}{N}\sum_{j=1}^{N} A_{ij}$, $\pi_i^C = \frac{\sum_{j=1}^{V} W_{ij}}{\sum_{l=1}^{N}\sum_{j=1}^{V} W_{lj}}$
4. **fin**
5. **répéter**

// Etape E de l'algorithme

6. $\quad$ **pour** $i = 1 \ldots N$, $j = 1 \ldots N$ et $k = 1 \ldots K$ **faire**
7. $\quad\quad \omega_{kij}^L \leftarrow \frac{\mu_{ki}\phi_{jk}^L}{\sum_{t=1}^{K}\mu_{ti}\phi_{jt}^L}$
8. $\quad$ **fin**
9. $\quad$ **pour** $i = 1 \ldots N$, $j = 1 \ldots V$ et $k = 1 \ldots K$ **faire**
10. $\quad\quad \omega_{kij}^C \leftarrow \frac{\mu_{ki}\phi_{jk}^C}{\sum_{t=1}^{K}\mu_{ti}\phi_{jt}^C}$
11. $\quad$ **fin**

// Etape M de l'algorithme

12. $\quad$ **pour** $i = 1 \ldots N$ et $k = 1 \ldots K$ **faire**
13. $\quad\quad \mu_{ki} = (1-\gamma) \times \frac{\alpha - 1 + \sum_{j=1}^{N} A_{ij}\omega_{kij}^L}{K(\alpha-1) + \sum_{t=1}^{K}\sum_{j=1}^{N} A_{ij}\omega_{tij}^L} + \gamma \times \frac{\sum_{j=1}^{V} W_{ij}\omega_{kij}^C}{\sum_{t=1}^{K}\sum_{j=1}^{V} W_{ij}\omega_{tij}^C}$
14. $\quad\quad \phi_{jk}^L = \frac{\beta - 1 + \sum_{i=1}^{N} A_{ij}\omega_{kij}^L}{N(\beta-1) + \sum_{l=1}^{N}\sum_{i=1}^{N} A_{il}\omega_{kil}^L}$; $\phi_{vk}^C = \frac{\sum_{i=1}^{N} W_{iv}\omega_{kiv}^C}{\sum_{l=1}^{V}\sum_{i=1}^{N} W_{il}\omega_{kil}^C}$
15. $\quad$ **fin**
16. **jusqu'à** *convergence*

fin

4.4.3 Mise en œuvre du modèle

Pour l'initialisation du modèle SPCE-PLSA, nous utilisons pour les paramètres $\boldsymbol{\varphi}^L$ une initialisation basée sur un clustering initial avec l'algorithme Graclus en utilisant la mesure de similarité d'Amsler. Pour les paramètres $\boldsymbol{\mu}$ et $\boldsymbol{\varphi}^C$, ils sont initialisés par des valeurs aléatoires.

Pour les paramètres de lissage α et β, nous adoptons la stratégie proposée dans la section 4.3.2 qui consiste à fixer ces paramètres aux valeurs suivantes :

$$\alpha = \beta = 1 + 10^{-1} \frac{M}{N \times K} \tag{4.20}$$

où M est le nombre total de liens dans le graphe de documents, N est le nombre de sommets (i.e. de documents) et K est le nombre de thèmes.

Enfin, le paramètre γ qui détermine l'importance des liens et des contenus a un impact important sur les performances du modèle SPCE-PLSA. Trouver la valeur optimale pour ce paramètre est une tâche délicate sur laquelle plusieurs chercheurs se sont penchés. A notre connaissance, il n'existe aucune solution satisfaisante dans la littérature à cette problématique. C'est pourquoi nous proposons d'utiliser, encore une fois, la méthode de la validation croisée en prenant comme critère la perplexité pour déterminer le meilleur compromis entre les liens et les contenus. Nous utilisons plus précisément une validation croisée à cinq passes où l'on divise les ensembles des liens et des occurrences de mots en cinq ensembles de tailles plus ou moins égales. L'apprentissage du modèle SPCE-PLSA se fait à chaque fois en utilisant 80% des données (i.e. des liens et des occurrences de mots) puis la perplexité est calculée pour les 20% restants. L'opération est répétée cinq fois et la perplexité finale est obtenue en faisant la moyenne des cinq valeurs de la perplexité obtenues à chaque étape de la validation croisée. Plus précisément, la perplexité est égale à la moyenne entre la perplexité par rapport aux liens et la perplexité par rapport aux contenus.

4.4.4 Résultats expérimentaux

Nous utilisons les corpus Cora et Citeseer pour évaluer les performances du modèle SPCE-PLSA en termes de NMI, de F-mesure, de modularité et de perplexité. Bien que la modularité ne soit pas un critère pertinent pour évaluer le regroupement en thématiques, nous reportons quand même les résultats avec cette mesure car elle montre l'impact de la prise en compte des contenus sur la structure de communautés identifiée. Notre modèle est comparé au modèle PHITS-PLSA. Les résultats obtenus sont indiqués par les figures 4.11, 4.12, 4.13 et 4.14. La perplexité avec le modèle PHITS-PLSA étant très élevée (de l'ordre de 10^{12}), nous n'avons pas rapporté les résultats qui la concernent.

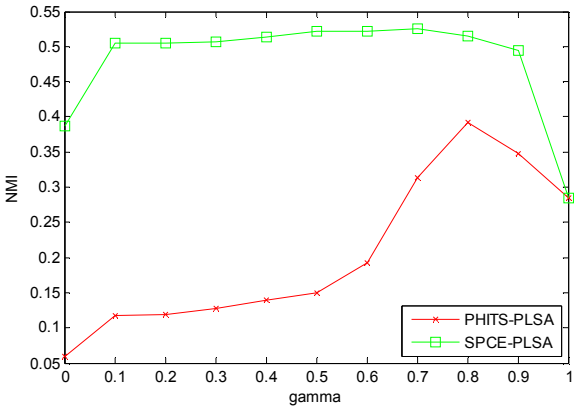


Figure 4.11 - La NMI en fonction du paramètre gamma pour le corpus Cora

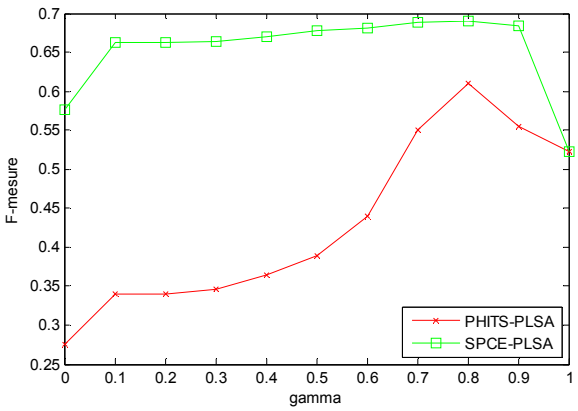


Figure 4.12 - La F-mesure en fonction du paramètre Gamma pour le corpus Cora

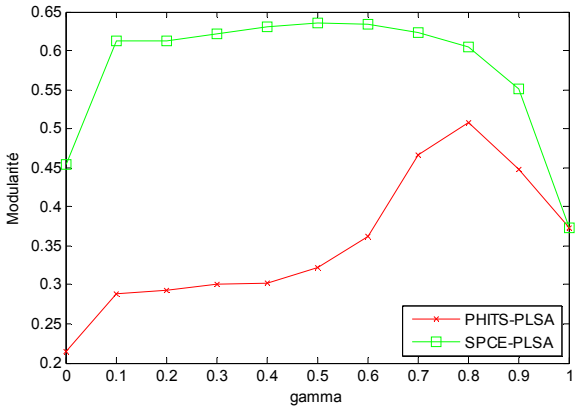


Figure 4.13 - La Modularité en fonction du paramètre gamma pour le corpus Cora

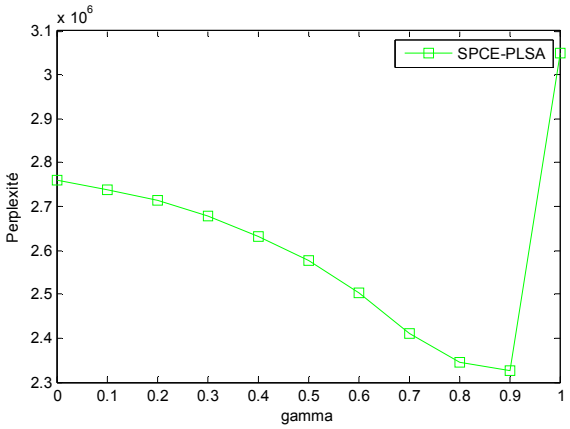


Figure 4.14 - La Perplexité en fonction du paramètre gamma pour le corpus Cora

Avec le corpus Cora, nous remarquons que l'utilisation combinée des liens et des contenus permet d'améliorer considérablement les performances en particulier avec le modèle SPCE-PLSA. Le paramètre gamma semble avoir beaucoup moins d'effet sur les performances de SPCE-PLSA que sur les performances de PHITS-PLSA. Pour SPCE-PLSA, il semblerait même que n'importe quelle valeur de gamma permet d'obtenir de meilleurs résultats qu'avec le modèle SPCE ou PLSA seuls. La perplexité indique par ailleurs que SPCE-PLSA explique mieux les données de test lorsque la valeur de gamma est proche de 1. Cela se justifie par le fait que la matrice documents/termes contient beaucoup plus de données que la matrice d'adjacence. Pour le corpus Cora par exemple, nous avons 5209 liens entre documents contre 155429 occurrences de mots dans le corpus, soit presque 30 fois plus que de liens. Enfin, la perplexité permet de déterminer une "bonne" valeur pour le paramètre gamma à savoir 0.9 pour le corpus Cora.

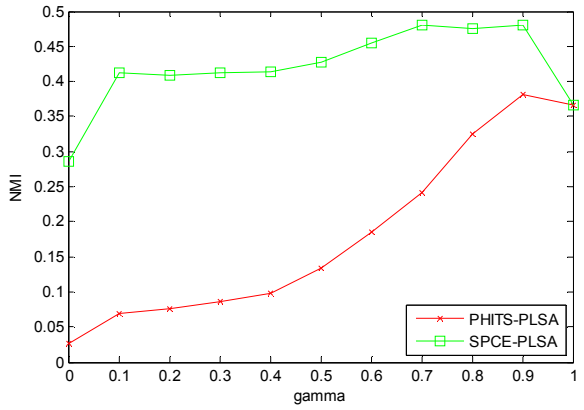


Figure 4.15 - La NMI en fonction du paramètre gamma pour le corpus Citeseer

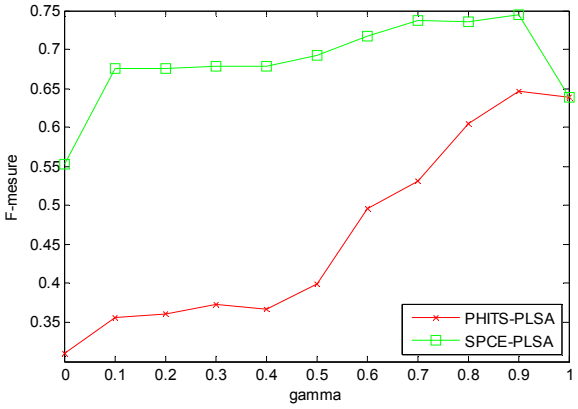


Figure 4.16 - La F-mesure en fonction du paramètre gamma pour le corpus Citeseer

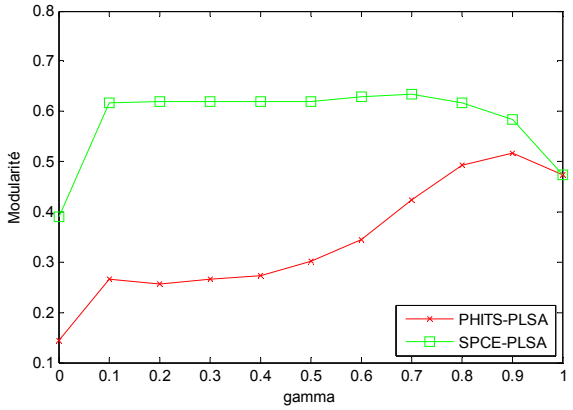


Figure 4.17 - La Modularité en fonction du paramètre gamma pour le corpus Citeseer

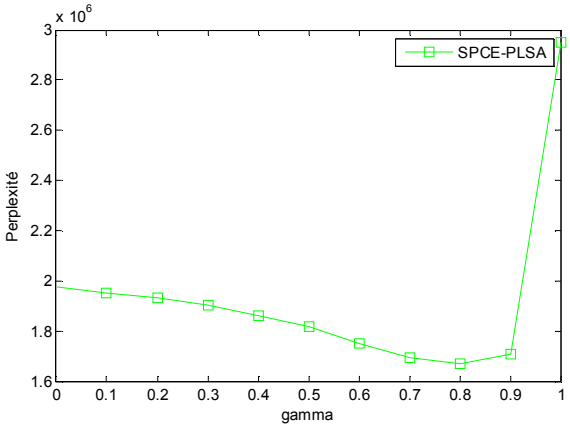


Figure 4.18 - La Perplexité en fonction du paramètre gamma pour le corpus Citeseer

Les figures 4.15, 4.16, 4.17 et 4.18 montrent que l'amélioration des performances en utilisant le modèle PHITS-PLSA avec le corpus Citeseer est moins bonne qu'avec le corpus Cora. La meilleure performance qu'atteint ce modèle (lorsque $\gamma = 0.9$) est en effet presque égale à la performance du modèle PLSA seul. PHITS ne semble pas apporter un plus pour l'identification de thématiques. Le modèle SPCE-PLSA, quant à lui, améliore considérablement les performances et ce quelle que soit la valeur du paramètre gamma. En d'autres termes, rien qu'en effectuant une analyse hybride avec SPCE-PLSA, on améliore l'identification de thématiques quelle que soit l'importance donnée aux liens et aux contenus. Les résultats de la perplexité indiquent que la meilleure valeur de celle-ci est atteinte lorsque $\gamma = 0.8$. Cette valeur est légèrement inférieure à la valeur précédente avec Cora (i.e. $\gamma = 0.9$) car le rapport entre le nombre d'occurrences de mots dans le corpus et le nombre total de liens entre documents est plus petit ($62506/3366 \simeq 19$).

4.5 Bilan

Nous avons proposé dans ce dernier chapitre le modèle génératif SPCE pour l'identification de structures de communautés dans les graphes de documents. Nous avons décrit son processus génératif, son algorithme d'estimation des paramètres, sa procédure d'initialisation ainsi qu'une méthode permettant à la fois de déterminer les valeurs optimales des paramètres de lissage et de calculer le nombre exact de communautés dans un graphe. Une partie expérimentale validant le modèle a notamment été présentée. Les résultats de celle-ci ont montré que le modèle SPCE possède de très bonnes performances.

En utilisant les critères que nous avons établis à la fin du troisième chapitre, nous résumons dans le tableau 4.5 les différentes propriétés du modèle SPCE.

Enfin, nous avons présenté le modèle SPCE-PLSA qui est un modèle génératif hybride combinant analyse de liens et analyse des contenus pour l'identification de thématiques i.e. pour la classification non supervisée de documents. L'évaluation du modèle SPCE-PLSA sur des corpus de documents a mis en valeur l'intérêt d'une approche hybride pour le regroupement de documents car celui-ci réalise de meilleures performances que le modèle SPCE ou le modèle PLSA appliqués séparément.

Propriété	**SPCE**
Identification de communautés dans des graphes orientés	oui
Identification de communautés qui se recouvrent	oui
Connaissances a priori	aucune
Complexité	$O(IKM)$
Déterministe	non
Identification de communautés de tailles et/ou de densités différentes	oui
Multi-vue sur la structure de communautés	oui

Tableau 4.5 – Propriétés du modèle SPCE
(*I* : nombre d'itérations, *K* : nombre de communautés, *M* : nombre total de liens)

Conclusion

Le travail réalisé dans le cadre de cette thèse s'inscrit dans le domaine de l'extraction de connaissances à partir de documents. Une des originalités de notre approche réside dans le fait d'analyser les liens entre documents au lieu d'analyser leurs contenus comme le font la plupart des travaux existants. Ce travail a été motivé par le besoin de caractériser de grandes collections de documents afin de faciliter leur utilisation et leur exploitation par des humains ou par des outils informatiques. Pour répondre à ce besoin, nous avons entrepris des recherches suivant deux axes qui, à première vue, peuvent sembler indépendants mais qui sont en fait liés. Concrètement, nous avons développé de nouveaux algorithmes d'analyse de liens entre documents en se basant sur des techniques d'apprentissage automatique.

Dans un premier temps, nous avons abordé la problématique du calcul de centralité dans les graphes de document. Il s'agit d'assigner un degré d'importance ou de popularité à chaque document en se basant uniquement sur ses connexions avec les autres documents du graphe. Dans ce premier volet de la thèse, nous avons décrit les principaux algorithmes de calcul de centralité existants en distinguant les approches issues de l'analyse des réseaux sociaux de celles issues de la recherche d'information. Nous avons montré que les mesures de centralité proposées récemment en recherche d'information (telles que HITS ou PageRank) correspondent en fait à des variantes de la centralité spectrale publiée au début des années 1970 par Bonacich. Nous avons également mis l'accent sur le problème TKC (Tightly Knit Community) dont souffre la plupart des mesures de centralité récentes. Ce problème est dû à la présence de communautés c'est-à-dire de sous-graphes de forte densité possédant peu de liens entre eux dans les graphes de documents (il s'agit en fait d'une propriété inhérente à ce type de graphes). En pratique, l'effet TKC se manifeste par une attribution non équitable des degrés d'importance aux différents documents. Ensuite, nous avons proposé trois nouveaux algorithmes de calcul de centralité dans les graphes de documents permettant d'affronter le phénomène TKC. Dans l'algorithme DocRank, un des trois algorithmes proposés, le degré d'autorité (resp. d'hubité) d'un document n'est pas simplement proportionnel au nombre de documents qui le citent (resp. qu'il cite) mais est plutôt égal à la somme des poids des recommandations qu'il reçoit (resp. qu'il effectue). Le poids de chaque recommandation est,

quant à lui, proportionnel ou inversement proportionnel (suivant un paramètre de normalisation) au degré sortant (resp. entrant) du nœud effectuant (resp. recevant) cette recommandation. DocRank possède entre autres l'avantage d'avoir une faible complexité puisqu'il ne nécessite aucun calcul de vecteurs propres contrairement aux algorithmes HITS, PageRank, etc. Les différents algorithmes proposés ont été évalués et comparés aux approches existantes (HITS, PageRank, SALSA, etc.) en utilisant huit graphes de documents. Des critères d'évaluation ont notamment été proposés pour mesurer l'effet TKC. Les résultats expérimentaux ont montré que l'algorithme DocRank donne les meilleurs résultats, c'est-à-dire que celui-ci est le moins affecté par l'effet TKC.

Dans un deuxième temps, nous nous sommes intéressés au problème de la classification non supervisée (ou regroupement automatique) de documents. Plus précisément, nous avons envisagé ce regroupement comme une tâche d'identification de structures de communautés (ISC) dans les graphes de documents. Les graphes de documents sont caractérisés par une faible densité globale ainsi que par un fort coefficient de regroupement ; ces deux caractéristiques indiquent en fait l'existence de communautés. La détection de telles structures dans les graphes a suscité ces dernières années l'attention d'un grand nombre de chercheurs qui ont proposé une panoplie d'approches pour l'ISC. Cependant, la plupart de ces techniques s'avèrent être peu adaptées à l'ISC dans les graphes de documents en raison de leurs particularités : l'orientation des liens, le recouvrement des communautés qu'ils contiennent et enfin leur "très" faible densité. Dans ce deuxième volet de la thèse, nous avons commencé par décrire les principales approches d'ISC existantes en distinguant les approches basées sur un modèle génératif (approches dites génératives) des approches algorithmiques ou classiques (approches dites non génératives). Nous avons ensuite proposé des critères permettant d'évaluer l'adéquation d'un algorithme d'ISC à l'analyse de graphes de documents. A l'issue du travail de synthèse réalisé, nous avons plaidé en faveur des approches génératives pour l'ISC dans les graphes de documents. Mais en pratique, nous avons remarqué que les modèles génératifs d'ISC existants donnent des résultats beaucoup moins satisfaisants que des approches classiques de clustering ou de partitionnement de graphes. Nous avons constaté que cette mauvaise performance est causée par la faible densité des graphes de documents. C'est pour cette raison que nous avons proposé un modèle génératif (SPCE) basé sur le lissage et sur une initialisation appropriée pour l'ISC dans des graphes de faible densité. Nous avons également étudié différentes solutions pour mettre en œuvre le modèle SPCE notamment en ce qui concerne la phase d'initialisation du modèle et l'estimation des paramètres de lissage. Le modèle SPCE a été évalué et validé en le comparant à d'autres approches génératives et non génératives d'ISC. L'évaluation expérimentale a été effectuée en utilisant quatre graphes de documents. Pour tous les graphes analysés, SPCE a obtenu de meilleures performances en particulier lorsqu'il est initialisé avec l'algorithme Graclus (en utilisant la mesure d'Amsler). En plus de ses très bonnes performances, le modèle SPCE possède les qualités suivantes :

- il est capable d'analyser des graphes orientés,
- il autorise le recouvrement des communautés,
- il détermine de manière automatique le nombre de communautés,

- il calcule simultanément deux regroupements (un à partir des liens entrants et l'autre à partir des liens sortants),
- il est générique et peut être utilisé pour l'ISC dans d'autres types de graphes,
- et surtout il possède une complexité linéaire par rapport au nombre total de liens dans le graphe.

Enfin, nous avons montré que le modèle SPCE pouvait être étendu pour prendre en compte les contenus des documents en plus des liens. Le modèle ainsi obtenu que nous avons appelé SPCE-PLSA (en référence au modèle PLSA d'analyse des contenues) a lui aussi été évalué en utilisant deux collections de documents. Les résultats expérimentaux ont montré que le regroupement de documents obtenu en exploitant simultanément les liens et les contenus est meilleur que celui obtenu en utilisant uniquement les liens ou les contenus. Les résultats de ce deuxième volet de la thèse illustrent, à notre avis, le grand intérêt d'une approche générative pour la classification non supervisée de documents.

Naturellement, de nombreuses perspectives sont envisageables pour poursuivre ce travail. Nous citons ci-dessous celles qui nous semblent les plus importantes.

Concernant le calcul de centralité, il serait, par exemple, intéressant de développer une technique permettant de déterminer de manière automatique la valeur optimale du paramètre de normalisation de l'algorithme DocRank. Lors de nos expérimentations, nous avons suggéré la valeur 1 pour ce paramètre qui correspond en fait à une version "démocratique" de DocRank, mais les résultats expérimentaux ont montré qu'une valeur supérieure à 1 permettait parfois d'obtenir de meilleurs résultats. Une autre piste de recherche concerne l'utilité de l'algorithme DocRank dans une tâche de recherche d'information. Nous envisageons en effet d'étudier l'apport de la mesure de centralité calculée par DocRank pour le calcul de la pertinence d'un document par rapport à une requête utilisateur. Dans ce contexte, une comparaison entre les différentes mesures de centralité serait également intéressante.

Quant à l'identification de structures de communautés, nous avons entamé le développement d'une version totalement bayésienne du modèle SPCE. Il s'agit d'une version dans laquelle tous les paramètres du modèle sont traités comme des variables dont la distribution doit être estimée. Une approche bayésienne possède l'avantage d'être moins sujette au problème du sur-apprentissage (overfitting) qui caractérise les approches fréquentistes (telles que SPCE ou PHITS). Elle offre également la possibilité de déterminer le nombre de communautés sans avoir à utiliser une méthode couteuse en temps de calcul telle que la validation croisée ; les approches bayésiennes permettent en effet de mieux quantifier la qualité d'un modèle probabiliste puisqu'elles prennent en compte à la fois la vraisemblance des données et la complexité du modèle. Enfin, pour le modèle hybride SPCE-PLSA proposé, nous envisageons d'étudier l'apport d'autres sources d'information telles que des ressources termino-ontologiques ou des tags pour la classification non supervisée de documents.

Annexe A
Eléments de la théorie des graphes

A.1 Graphes non orientés

Un **graphe non orienté** $G=(V,E)$ est défini par un ensemble de *sommets* V (appelés aussi nœuds) et un ensemble d'*arêtes* E qui relient ces sommets. $N=|V|$ est le nombre de sommets du graphe appelé aussi *ordre* du graphe. $M=|E|$ est le nombre d'arêtes du graphe appelé aussi *taille* du graphe.

Deux nœuds reliés par une arête sont dits **adjacents**.

La **matrice d'adjacence** d'un graphe $G=(V,E)$ d'ordre N est une matrice carrée de dimension $N\times N$ telle que l'entrée (i,j) de cette matrice soit égale à 1 si les nœuds *i* et *j* sont reliés par une arête et 0 sinon.

Le **voisinage** d'un nœud correspond à l'ensemble de tous ses nœuds adjacents.

Le **degré** d'un sommet est le nombre de sommets qui lui sont adjacents.

Un nœud de degré zéro est dit nœud **isolé**. Un nœud de degré 1 est appelé **feuille** ou nœud pendant.

Une **chaîne** est une suite finie d'arêtes consécutives (i.e. qui ont une extrémité en commun). La *longueur* d'une chaîne est égale au nombre d'arêtes qui la composent.

La **distance** entre deux sommets est la longueur de la plus courte chaîne entre ces sommets ; elle est appelée aussi *distance géodésique*.

Le **diamètre** d'un graphe est la plus grande distance entre deux sommets de ce graphe.

Un graphe $S=(V_S,E_S)$ est un **sous-graphe** de $G=(V,E)$ si $V_S\subset V$ et $E_S\subset E$.

Dans un graphe **complet**, chaque sommet est adjacent à tous les autres sommets du graphe.

Un graphe est **connexe** s'il existe une chaîne entre toute paire de nœuds.

Une **composante connexe** est un sous-graphe connexe maximal.

A.2 Graphes orientés

Un **graphe orienté** (ou *digraphe*) est un graphe où les arêtes sont orientées et sont appelées des *arcs*.

Le **degré entrant** d'un sommet *i* est le nombre d'arcs dont l'extrémité est *i*.

Le **degré sortant** d'un sommet *i* est le nombre d'arcs dont l'origine est *i*.

Le **degré total** d'un sommet est égal à la somme des degrés entrant et sortant de ce nœud.

Un **chemin** est une suite finie d'arcs consécutifs (i.e. l'extrémité de chaque arc coïncide avec l'origine de l'arc suivant). La *longueur* d'un chemin est égale au nombre d'arcs qui le composent.

La **distance** entre deux sommets est la longueur du plus court chemin entre ces sommets.

Un graphe orienté est dit **fortement connexe** s'il existe un chemin entre tous les couples de nœuds du graphe.

Un graphe orienté est dit **faiblement connexe** s'il existe une chaîne entre tous les couples de nœuds du graphe, autrement dit si sa version non orientée est connexe.

Annexe B
Compléments au chapitre 2

B.1 Résultats avec le graphe "Apple"

0.0097165 http://www.tuaw.com In:351 Out:157
0.0088307 http://www.engadget.com In:319 Out:130
0.0077511 http://www.apple.com In:280 Out:0
0.0077234 http://www.joystiq.com In:279 Out:105
0.0077234 http://www.xbox360fanboy.com In:279 Out:120
0.007585 http://www.downloadsquad.com In:274 Out:116
0.0075019 http://www.secondlifeinsider.com In:271 Out:121
0.0074743 http://www.pspfanboy.com In:270 Out:116
0.0072251 http://www.engadgetmobile.com In:261 Out:124
0.0072251 http://www.slashfood.com In:261 Out:116

Figure B.1 – Liste des 10 meilleures autorités retournée par DEG

0.0043461 http://www.tuaw.com In:351 Out:157
0.004097 http://www.tuaw.com/category/apple In:32 Out:148
0.0039309 http://www.engadget.com/tag/apple In:25 Out:142
0.0038755 http://engadget.com/tag/iphone In:0 Out:140
0.0037371 http://cellphones.engadget.com In:1 Out:135
0.0037371 http://www.tuaw.com/category/apple-financial In:6 Out:135
0.0036818 http://www.tuaw.com/2007/07/11/tuaw-interview-visto-corporate-email-for-iphone In:162 Out:133
0.0036541 http://www.tuaw.com/category/humor In:8 Out:132
0.0035987 http://www.engadget.com/2006/08/24/apple-to-recall-1-8-million-sony-made-batteries In:23 Out:130
0.0035987 http://www.tuaw.com/2006/08/26/cpsc-and-apple-get-recall-battery-lists-synced In:0 Out:130

Figure B.2 – Liste des 10 meilleurs hubs retournée par DEG

0.0044393 http://www.cinematical.com/2007/07/11/zachary-quinto-in-talks-to-... In:119 Out:126
0.0044393 http://www.cinematical.com/2007/07/11/singer-will-do-superman-...In:119 Out:126
0.0044393 http://www.cinematical.com In:254 Out:126
0.0044393 http://www.cinematical.com/2007/07/11/poster-for-julia-roberts-fireflies-in-the-garden In:119 Out:126
0.0044393 http://www.cinematical.com/2007/07/11/alex-proyas-will-direct-dracula-year-zero In:119 Out:126
0.0044393 http://www.cinematical.com/2007/07/11/alec-baldwin-please-dont-go-see-my-new-movie In:119 Out:126
0.0044391 http://www.adjab.com/2005/04/30/compendium-of-apple-advertising In:0 Out:127
0.0044385 http://www.adjab.com In:259 Out:126
0.0044384 http://www.bloggingneworleans.com In:250 Out:126
0.0043417 http://www.autoblog.com/2007/07/11/manhattan-project-proposal-seeks-to-shut-down-nyc-traffic In:187 Out:125

Figure B.3 – Liste des 10 meilleurs hubs retournée par HITS

0.0097165 http://www.tuaw.com In:351 Out:157
0.0088307 http://www.engadget.com In:319 Out:130
0.0077511 http://www.apple.com In:280 Out:0
0.0077234 http://www.joystiq.com In:279 Out:105
0.0077234 http://www.xbox360fanboy.com In:279 Out:120
0.007585 http://www.downloadsquad.com In:274 Out:116
0.0075019 http://www.secondlifeinsider.com In:271 Out:121
0.0074743 http://www.pspfanboy.com In:270 Out:116
0.0072251 http://www.engadgetmobile.com In:261 Out:124
0.0072251 http://www.slashfood.com In:261 Out:116

Figure B.4 – Liste des 10 meilleures autorités retournée par SALSA

0.0043461 http://www.tuaw.com In:351 Out:157
0.004097 http://www.tuaw.com/category/apple In:32 Out:148
0.0039309 http://www.engadget.com/tag/apple In:25 Out:142
0.0038755 http://engadget.com/tag/iphone In:0 Out:140
0.0037371 http://cellphones.engadget.com In:1 Out:135
0.0037371 http://www.tuaw.com/category/apple-financial In:6 Out:135
0.0036818 http://www.tuaw.com/2007/07/11/tuaw-interview... In:162 Out:133
0.0036541 http://www.tuaw.com/category/humor In:8 Out:132
0.0035987 http://www.engadget.com/2006/08/24/apple-to-recall-... In:23 Out:130
0.0035987 http://www.tuaw.com/2006/08/26/cpsc-and-apple-get-... In:0 Out:130

Figure B.5 – Liste des 10 meilleurs hubs retournée par SALSA

0.0096813 http://www.xbox360fanboy.com In:279 Out:120
0.0095998 http://www.secondlifeinsider.com In:271 Out:121
0.0094641 http://www.pspfanboy.com In:270 Out:116
0.0093977 http://www.engadget.com In:319 Out:130
0.0092477 http://www.joystiq.com In:279 Out:105
0.0092095 http://www.downloadsquad.com In:274 Out:116
0.0091322 http://www.tuaw.com In:351 Out:157
0.009115 http://www.engadgetmobile.com In:261 Out:124
0.0091139 http://www.slashfood.com In:261 Out:116
0.0090268 http://www.adjab.com In:259 Out:126

Figure B.6 – Liste des 10 meilleures autorités retournée par HubAvg

0.0033709
http://www.technorati.com/cosmos/search.html?rank=&fc=1&url=http://www.tuaw.com/2007/06/29/apple-store-online-down In:1 Out:3
0.0033415
http://www.technorati.com/cosmos/search.html?rank=&fc=1&url=http://www.tuaw.com/2007/07/10/found-footage-iphone-dev-camp-hackathon In:1 Out:4
0.0033145
http://www.technorati.com/cosmos/search.html?rank=&fc=1&url=http://www.tuaw.com/2007/07/10/adium-x-hits-1-0-5 In:1 Out:5
0.0033061 http://bookoftech.blogspot.com In:1 Out:1
0.0033061 http://popsci.typepad.com/popsci/2007/06/gotta-get-an-ip.html In:1 Out:1
0.0032943
http://www.technorati.com/cosmos/search.html?rank=&fc=1&url=http://www.tuaw.com/2007/06/29/launch-gallery-chicago-il In:1 Out:6
0.0032816
http://www.technorati.com/cosmos/search.html?rank=&fc=1&url=http://www.tuaw.com/2007/07/08/spy-shot-apple-store-touchwood-uk In:1 Out:6
0.0032773
http://www.technorati.com/cosmos/search.html?rank=&fc=1&url=http://www.tuaw.com/2007/06/29/line-report-iphone-mania In:1 Out:7
0.0032773
http://www.technorati.com/cosmos/search.html?rank=&fc=1&url=http://www.tuaw.com/2007/06/29/it-aint-easy-to-get-an-iphone-review-unit In:1 Out:7
0.0032728
http://www.technorati.com/cosmos/search.html?rank=&fc=1&url=http://www.tuaw.com/2007/06/29/line-update-sherman-oaks-ca-apple-store In:1 Out:6

Figure B.7 – Liste des 10 meilleurs hubs retournée par HubAvg

0.008703 http://www.xbox360fanboy.com In:279 Out:120
0.0086658 http://www.secondlifeinsider.com In:271 Out:121
0.0086624 http://www.cssinsider.com In:251 Out:47
0.0086398 http://www.dsfanboy.com In:257 Out:121
0.0086389 http://www.bbhub.com In:253 Out:116
0.0086366 http://www.pspfanboy.com In:270 Out:116
0.0086347 http://www.ps3fanboy.com In:254 Out:116
0.0086339 http://www.nintendowiifanboy.com In:254 Out:121
0.0086286 http://www.wowinsider.com In:252 Out:116
0.0086276 http://www.droxy.com In:251 Out:121

Figure B.8 – Liste des 10 meilleures autorités retournée par MHITS

0.0044287 http://www.cinematical.com/2007/07/11/zachary-quinto-... In:119 Out:126
0.0044287 http://www.cinematical.com/2007/07/11/singer-will-do-... In:119 Out:126
0.0044287 http://www.cinematical.com In:254 Out:126
0.0044287 http://www.cinematical.com/2007/07/11/poster-for-julia-... In:119 Out:126
0.0044287 http://www.cinematical.com/2007/07/11/alex-proyas-will-... In:119 Out:126
0.0044287 http://www.cinematical.com/2007/07/11/alec-baldwin-... In:119 Out:126
0.0044285 http://www.adjab.com/2005/04/30/compendium-of-apple-... In:0 Out:127
0.004428 http://www.adjab.com In:259 Out:126
0.0044279 http://www.bloggingneworleans.com In:250 Out:126
0.0043314 http://www.autoblog.com/2007/07/11/honda-to-bring-... In:187 Out:125

Figure B.9 – Liste des 10 meilleurs hubs retournée par MHITS

0.0088525 http://www.engadget.com In:319 Out:130
0.0086075 http://www.joystiq.com In:279 Out:105
0.0084721 http://www.tuaw.com In:351 Out:157
0.0084088 http://www.xbox360fanboy.com In:279 Out:120
0.0082472 http://www.downloadsquad.com In:274 Out:116
0.0081567 http://www.secondlifeinsider.com In:271 Out:121
0.0080967 http://www.pspfanboy.com In:270 Out:116
0.008054 http://ted.aol.com In:247 Out:4
0.0080447 http://www.dsfanboy.com In:257 Out:121
0.0080299 http://www.cssinsider.com In:251 Out:47

Figure B.10 – Liste des 10 meilleures autorités retournée par NHITS_10

0.0041414 http://www.tuaw.com In:351 Out:157
0.004129 http://www.tuaw.com/category/apple-financial In:6 Out:135
0.0040893 http://www.tuaw.com/category/humor In:8 Out:132
0.0040885 http://www.tuaw.com/category/flickr-find In:0 Out:129
0.0040856 http://www.tuaw.com/2007/01/22/flickr-find-things-beside-steve-that-go-boom In:0 Out:128
0.0040848 http://www.tuaw.com/category/apple In:32 Out:148
0.0040815 http://www.tuaw.com/2004/11/16/ipod-photo-renamed-now-called-ipod-photo In:0 Out:127
0.0040799 http://www.tuaw.com/2006/10/26/macexpo-photo-gallery-and-greenpeace-goes-after-apples-iwaste In:0 Out:128
0.0040783 http://www.tuaw.com/category/cult-of-mac In:7 Out:127
0.0040753 http://www.tuaw.com/2007/07/11/tuaw-interview-visto-corporate-email-for-iphone In:162 Out:133

Figure B.11 – Liste des 10 meilleurs hubs retournée par NHITS_10

0.011557 http://www.tuaw.com In:351 Out:157
0.0094591 http://www.xbox360fanboy.com In:279 Out:120
0.0089062 http://www.downloadsquad.com In:274 Out:116
0.0088249 http://www.secondlifeinsider.com In:271 Out:121
0.0087858 http://www.joystiq.com In:279 Out:105
0.0087821 http://www.pspfanboy.com In:270 Out:116
0.0087632 http://www.engadget.com In:319 Out:130
0.0087215 http://www.engadgetmobile.com In:261 Out:124
0.0084908 http://www.slashfood.com In:261 Out:116
0.0078831 http://www.adjab.com In:259 Out:126

Figure B.12 – Liste des 10 meilleures autorités retournée par NHITS_20

0.003722 http://engadget.com/tag/iphone In:0 Out:140
0.0037191 http://www.engadget.com/2006/04/01/30-years-in-apple-products-the-good-the-bad-and-the-ugly In:4 Out:128
0.0037181 http://cellphones.engadget.com In:1 Out:135
0.0037135 http://www.engadget.com/tag/apple In:25 Out:142
0.0036888 http://www.engadget.com/2006/08/24/apple-to-recall-1-8-million-sony-made-batteries In:23 Out:130
0.003683 http://www.engadget.com/2007/02/03/ce-oh-no-he-didnt-part-xxiii-gates-security-guys-break-the In:0 Out:127
0.0036817 http://www.engadget.com In:319 Out:130
0.0036785 http://www.engadget.com/2007/07/11/macbook-pro-12-inch-ultraportable-rumor-resurfaces In:174 Out:128
0.0036783 http://www.engadget.com/tag/iBook In:0 Out:128
0.0036781 http://wireless.engadget.com In:222 Out:128

Figure B.13 – Liste des 10 meilleurs hubs retournée par NHITS_20

0.0085065 http://www.xbox360fanboy.com In:279 Out:120
0.0084391 http://www.cssinsider.com In:251 Out:47
0.0084367 http://www.secondlifeinsider.com In:271 Out:121
0.0084189 http://www.dsfanboy.com In:257 Out:121
0.0084174 http://www.bbhub.com In:253 Out:116
0.0084171 http://www.pspfanboy.com In:270 Out:116
0.0084156 http://www.ps3fanboy.com In:254 Out:116
0.0084141 http://www.nintendowiifanboy.com In:254 Out:121
0.0084132 http://www.wowinsider.com In:252 Out:116
0.0084057 http://www.droxy.com In:251 Out:121

Figure B.14 – Liste des 10 meilleures autorités retournée par DocRank_0

0.0041708 http://www.adjab.com/2005/04/30/compendium-of-apple-advertising In:0 Out:127
0.0041686 http://www.bloggingneworleans.com In:250 Out:126
0.0041679 http://www.cinematical.com In:254 Out:126
0.0041679 http://www.cinematical.com/2007/07/11/poster-for-julia-roberts-fireflies-in-the-garden In:119 Out:126
0.0041679 http://www.cinematical.com/2007/07/11/zachary-quinto-in-talks-to-play-spock-source-says In:119 Out:126
0.0041679 http://www.cinematical.com/2007/07/11/alex-proyas-will-direct-dracula-year-zero In:119 Out:126
0.0041679 http://www.cinematical.com/2007/07/11/alec-baldwin-please-dont-go-see-my-new-movie In:119 Out:126
0.0041679 http://www.cinematical.com/2007/07/11/singer-will-do-superman-sequel-with-kevin-spacey-returning-b In:119 Out:126
0.0041672 http://www.adjab.com In:259 Out:126
0.0040828 http://www.autoblog.com In:251 Out:125

Figure B.15 – Liste des 10 meilleurs hubs retournée par DocRank_0

0.05522 http://www.apple.com In:280 Out:0
0.030964 http://www.download.com/itunes-for-windows/3000-2166_4-10235268.html In:46 Out:12
0.029992 http://www.mac.com In:57 Out:1
0.016984 http://developer.apple.com/wwdc In:84 Out:0
0.015538 http://store.apple.com In:47 Out:0
0.015503 http://www.tuaw.com In:351 Out:157
0.014977 http://www.apple-history.com In:114 Out:0
0.014121 http://projects.info-pull.com/moab In:32 Out:13
0.013732 http://www.apple.com/iphone In:69 Out:0
0.012866 http://techrepublic.com.com In:34 Out:1

Figure B.16 – Liste des 10 meilleures autorités retournée par DocRank_1.5

0.050973 http://en.wikipedia.org/wiki/Apple_Computer In:12 Out:72
0.045345 http://www.tuaw.com/category/apple In:32 Out:148
0.04279 http://www.answers.com/topic/apple-computer-inc In:11 Out:71
0.029882 http://www.techcrunch.com/2006/09/29/why-the-new-mac-webmail-is-important In:27 Out:16
0.029318 http://de.wikipedia.org/wiki/Apple In:11 Out:44
0.029313 http://blogs.zdnet.com/Apple In:30 Out:29
0.028195 http://projects.info-pull.com/moab In:32 Out:13
0.027171 http://www.tuaw.com In:351 Out:157
0.026326 http://www.computerbase.de/lexikon/Apple In:0 Out:44
0.016877 http://apple.corante.com In:21 Out:29

Figure B.17 – Liste des 10 meilleurs hubs retournée par DocRank_1.5

B.2 Résultats avec le graphe "Armstrong"

0.021299 http://www.satchmo.net In:100 Out:4
0.017678 http://www.redhotjazz.com/louie.html In:83 Out:0
0.017039 http://www.lancearmstrong.com In:80 Out:2
0.014483 http://en.wikipedia.org/wiki/Neil_Armstrong In:68 Out:36
0.013845 http://www.jsc.nasa.gov/Bios/htmlbios/armstrong-na.html In:65 Out:0
0.013632 http://starchild.gsfc.nasa.gov/docs/StarChild/whos_who_... In:64 Out:0
0.013632 http://www.npg.si.edu/exh/armstrong In:64 Out:0
0.013632 http://www.livestrong.org In:64 Out:2
0.013419 http://www.armstrong.com In:63 Out:3
0.012993 http://en.wikipedia.org/wiki/Louis_Armstrong In:61 Out:23

Figure B.18 – Liste des 10 meilleures autorités retournée par DEG

0.010437 http://www.wunderground.com/global/stations/71841.html In:11 Out:49
0.0076677 http://en.wikipedia.org/wiki/Neil_Armstrong In:68 Out:36
0.0076677 http://en.wikipedia.org/wiki/Neil_A._Armstrong In:0 Out:36
0.0076677 http://scores.espn.go.com/nhl/recap?gameId=270106008 In:0 Out:36
0.0076677 http://scores.espn.go.com/nhl/recap?gameId=270206020 In:0 Out:36
0.0076677 http://espndeportes.espn.go.com/nhl/recap?gameId=261130024 In:0 Out:36
0.0076677 http://espndeportes.espn.go.com/nhl/recap?gameId=240331008 In:0 Out:36
0.0074547 http://sports.espn.go.com/nhl/players/profile?statsId=1029 In:18 Out:35
0.0066028 http://www.answers.com/topic/neil-armstrong In:2 Out:31
0.0051118 http://www.samairearmstrong.net In:52 Out:24

Figure B.19 – Liste des 10 meilleurs hubs retournée par DEG

0.14311 http://espndeportes.espn.go.com/nhl/recap?gameId=240331008 In:0 Out:36
0.14311 http://espndeportes.espn.go.com/nhl/recap?gameId=261130024 In:0 Out:36
0.14311 http://scores.espn.go.com/nhl/recap?gameId=270106008 In:0 Out:36
0.14311 http://scores.espn.go.com/nhl/recap?gameId=270206020 In:0 Out:36
0.1398 http://sports.espn.go.com/nhl/players/profile?statsId=1029 In:18 Out:35
0.066994 http://www.active.com/espn/getactive In:5 Out:16
0.0044766 http://www.brainerdhockey.com/links.htm In:0 Out:3
0.0044766 http://www.letsplayhockey.com/sponsors.html In:0 Out:3
0.0044766 http://www.hockeyzonemn.com/links.html In:0 Out:3
0.0044745 http://shoes.mu.nu In:0 Out:2

Figure B.20 – Liste des 10 meilleurs hubs retournée par HITS

0.23522 http://www.satchmo.net In:100 Out:4
0.1559 http://www.redhotjazz.com/louie.html In:83 Out:0
0.097278 http://www.npg.si.edu/exh/armstrong In:64 Out:0
0.06989 http://www.pbs.org/jazz/biography/artist_id_armstrong_louis.htm In:57 Out:3
0.064158 http://www.time.com/time/time100/artists/profile/armstrong.html In:52 Out:5
0.031762 http://en.wikipedia.org/wiki/Louis_Armstrong In:61 Out:23
0.028138 http://www.cosmopolis.ch/cosmo19/armstrong.htm In:50 Out:6
0.02384 http://www.pbs.org/wnet/americanmasters/database/armstrong_l.html In:40 Out:4
0.019444 http://www.satchography.com In:17 Out:0
0.018399 http://www.armstrong.com In:63 Out:3

Figure B.21 – Liste des 10 meilleures autorités retournée par HubAvg

0.0056793 http://www.oconnormusic.org/music.htm In:0 Out:1
0.0056793 http://www.myfavouritesongs.com In:0 Out:1
0.0056793 http://www.oldiesmusic.com/links.htm In:0 Out:1
0.0056793 http://www.normangeras.blogspot.com/2003_07_27_... In:0 Out:1
0.0056793 http://www.butlerwebs.com/holidays/august.htm In:0 Out:1
0.0056793 http://www.butlerwebs.com/holidays/july.htm In:0 Out:1
0.0056793 http://www.jazzsingers.com/SingersSites In:0 Out:1
0.0056793 http://www.december.com/places/nyc/all.html In:0 Out:1
0.0056793 http://www.hightechscience.org/museums.htm In:0 Out:1
0.0056793 http://www.cb3qn.nyc.gov In:0 Out:1

Figure B.22 – Liste des 10 meilleurs hubs retournée par HubAvg

0.038918 http://www.satchmo.net In:100 Out:4
0.035213 http://www.redhotjazz.com/louie.html In:83 Out:0
0.029469 http://www.lancearmstrong.com In:80 Out:2
0.025258 http://www.livestrong.org In:64 Out:2
0.021404 http://en.wikipedia.org/wiki/Neil_Armstrong In:68 Out:36
0.021394 http://www.npg.si.edu/exh/armstrong In:64 Out:0
0.020827 http://starchild.gsfc.nasa.gov/docs/StarChild/whos_who... In:64 Out:0
0.020287 http://www.mediawiki.org In:56 Out:1
0.020129 http://www.jsc.nasa.gov/Bios/htmlbios/armstrong-na.html In:65 Out:0
0.01927 http://wikimediafoundation.org In:49 Out:1

Figure B.23 – Liste des 10 meilleures autorités retournée par NHITS_10

0.021388 http://espndeportes.espn.go.com/nhl/recap?gameId=240331008 In:0 Out:36
0.021388 http://scores.espn.go.com/nhl/recap?gameId=270106008 In:0 Out:36
0.021388 http://scores.espn.go.com/nhl/recap?gameId=270206020 In:0 Out:36
0.021388 http://espndeportes.espn.go.com/nhl/recap?gameId=261130024 In:0 Out:36
0.020892 http://sports.espn.go.com/nhl/players/profile?statsId=1029 In:18 Out:35
0.014946 http://en.wikipedia.org/wiki/Neil_A._Armstrong In:0 Out:36
0.014946 http://en.wikipedia.org/wiki/Neil_Armstrong In:68 Out:36
0.012618 http://www.answers.com/topic/neil-armstrong In:2 Out:31
0.010016 http://www.active.com/espn/getactive In:5 Out:16
0.0056383 http://www.fibsarmstrong.info In:0 Out:24

Figure B.24 – Liste des 10 meilleurs hubs retournée par NHITS_10

0.025973 http://www.satchmo.net In:100 Out:4
0.024896 http://en.wikipedia.org/wiki/Louis_Armstrong In:61 Out:23
0.024457 http://www.armstrong.com In:63 Out:3
0.023921 http://www.redhotjazz.com/louie.html In:83 Out:0
0.022698 http://www.lancearmstrong.com In:80 Out:2
0.021989 http://www.armstrongcounty.com In:58 Out:2
0.021774 http://www.livestrong.org In:64 Out:2
0.019849 http://en.wikipedia.org/wiki/Neil_Armstrong In:68 Out:36
0.016412 http://www.armstrong.org In:60 Out:7
0.016405 http://www.co.armstrong.pa.us In:53 Out:0

Figure B.25 – Liste des 10 meilleures autorités retournée par NHITS_20

0.012246 http://espndeportes.espn.go.com/nhl/recap?gameId=240331008 In:0 Out:36
0.012246 http://espndeportes.espn.go.com/nhl/recap?gameId=261130024 In:0 Out:36
0.012246 http://scores.espn.go.com/nhl/recap?gameId=270206020 In:0 Out:36
0.012246 http://scores.espn.go.com/nhl/recap?gameId=270106008 In:0 Out:36
0.011962 http://sports.espn.go.com/nhl/players/profile?statsId=1029 In:18 Out:35
0.011212 http://en.wikipedia.org/wiki/Neil_A._Armstrong In:0 Out:36
0.011212 http://en.wikipedia.org/wiki/Neil_Armstrong In:68 Out:36
0.0096862 http://www.answers.com/topic/neil-armstrong In:2 Out:31
0.005979 http://www.fibsarmstrong.info In:0 Out:24
0.0057431 http://www.active.com/espn/getactive In:5 Out:16

Figure B.26 – Liste des 10 meilleurs hubs retournée par NHITS_20

0.013924 http://www.satchmo.net In:100 Out:4
0.012801 http://www.mediawiki.org In:56 Out:1
0.012427 http://wikimediafoundation.org In:49 Out:1
0.011715 http://wikimediafoundation.org/wiki/Fundraising In:34 Out:1
0.010443 http://www.lancearmstrong.com In:80 Out:2
0.0098065 http://www.jsc.nasa.gov/Bios/htmlbios/armstrong-na.html In:65 Out:0
0.0093573 http://www.redhotjazz.com/louie.html In:83 Out:0
0.0085339 http://en.wikipedia.org/wiki/Neil_Armstrong In:68 Out:36
0.0077853 http://www.nhl.com In:11 Out:0
0.0077104 http://www.velonews.com In:16 Out:0

Figure B.27 – Liste des 10 meilleures autorités retournée par DocRank_0

0.0067145 http://www.fibsarmstrong.info In:0 Out:24
0.0050359 http://www.smile2find.info/armstrong.htm In:0 Out:15
0.0046091 http://www.ryanmusicstore.com/1/musicstore43.html In:0 Out:14
0.0045579 http://www.worldwidirectory.com/North.America/... In:0 Out:14
0.0042677 http://www.worldwidirectory.com/North.America/United.States/... In:0 Out:13
0.0031467 http://www.usatraveltrip.com/armstrong.html In:0 Out:9
0.00305 http://www.webs-de-personajes.com.ar/buscar.php?... In:0 Out:9
0.002407 http://www.hardwoodflooringab.com/armstrongflooringcanada In:0 Out:7
0.0022761 http://www.brothersjudd.com/index.cfm/fuseaction/... In:0 Out:8
0.0021111 http://www.flooringab.com/armstronglaminantflooring In:0 Out:7

Figure B.28 – Liste des 10 meilleurs hubs retournée par DocRank_0

0.074503 http://www.wunderground.com/global/stations/71841.html In:11 Out:49
0.025261 http://www.samairearmstrong.net In:52 Out:24
0.019647 http://www.careerbuilder.com/company/jobs/c/... In:4 Out:21
0.016735 http://en.wikipedia.org/wiki/Neil_Armstrong In:68 Out:36
0.016735 http://en.wikipedia.org/wiki/Neil_A._Armstrong In:0 Out:36
0.016573 http://www.vh1.com/artists/az/armstrong_louis/artist.jhtml In:5 Out:11
0.016556 http://www.sensesofcinema.com/contents/directors/02/... In:24 Out:10
0.015847 http://space.about.com/od/astronautbiographies/a/neilarmstrong.htm In:23 Out:16
0.014947 http://www.answers.com/topic/neil-armstrong In:2 Out:31
0.014803 http://www.answers.com/topic/louis-armstrong In:18 Out:22

Figure B.29 – Liste des 10 meilleurs hubs retournée par DocRank_1

0.019705 http://www.armstrongcounty.com In:58 Out:2
0.019387 http://www.joearmstrong.com In:50 Out:1
0.019356 http://www.armstronggarden.com In:51 Out:1
0.019312 http://www.armstrongmold.com In:50 Out:0
0.018981 http://www.armstrongtools.com In:51 Out:0
0.018883 http://www.armstrong.com In:63 Out:3
0.01849 http://www.armstrongglass.com In:52 Out:4
0.018488 http://www.salon.com/books/int/2006/05/30/armstrong In:50 Out:2
0.018433 http://www.armstrong-ceilings.co.uk In:51 Out:0
0.018324 http://users.erols.com/oldradio/index.htm In:50 Out:0

Figure B.30 – Liste des 10 meilleures autorités retournée par DocRank_1.5

0.12513 http://www.wunderground.com/global/stations/71841.html In:11 Out:49
0.03801 http://www.samairearmstrong.net In:52 Out:24
0.028876 http://www.vh1.com/artists/az/armstrong_louis/artist.jhtml In:5 Out:11
0.028875 http://www.sensesofcinema.com/contents/directors/02/... In:24 Out:10
0.024579 http://www.careerbuilder.com/company/jobs/c/... In:4 Out:21
0.0231 http://sportsillustrated.cnn.com/more/specials/tour_de_france/2004 In:23 Out:8
0.022032 http://space.about.com/od/astronautbiographies/a/neilarmstrong.htm In:23 Out:16
0.021291 http://www.usatoday.com/sports/cycling/tourdefrance/... In:12 Out:11
0.020446 http://www.armstronghockey.com In:53 Out:10
0.020213 http://www.armstrong.org In:60 Out:7

Figure B.31 – Liste des 10 meilleurs hubs retournée par DocRank_1.5

B.3 Résultats avec le graphe "Jaguar "

0.012454 http://www.ps3fanboy.com In:169 Out:133
0.012036 http://www.nintendowiifanboy.com In:169 Out:143
0.011286 http://www.jaguar.com In:119 Out:0
0.010892 http://www.jag-lovers.org In:146 Out:4
0.010803 http://www.cardsquad.com In:169 Out:138
0.010688 http://www.scmwire.com In:169 Out:169
0.010661 http://www.blogginggnomedex.com In:169 Out:167
0.010304 http://www.bbhub.com In:169 Out:138
0.0102 http://www.secondlifeinsider.com In:169 Out:143
0.0099161 http://www.thediabetesblog.com In:170 Out:133

Figure B.32 – Liste des 10 meilleures autorités retournée par DEG

0.0084653 http://www.autoblog.com/category/jaguar In:19 Out:201
0.0074623 http://pages.citebite.com/m9u1y2q6voja In:0 Out:115
0.006852 http://e3.weblogsinc.com In:130 Out:128
0.0068508 http://www.engadget.com In:171 Out:142
0.0067249 http://www.bloggingohio.com In:169 Out:148
0.0065388 http://www.slashfood.com In:165 Out:133
0.0065048 http://bluetooth.weblogsinc.com In:130 Out:128
0.0064342 http://automoviles.aol.com In:67 Out:103
0.0062614 http://digitalmusic.weblogsinc.com In:130 Out:107
0.0062518 http://gps.engadget.com In:161 Out:142

Figure B.33 – Liste des 10 meilleurs hubs retournée par DEG

0.013565 http://www.bloggingohio.com In:169 Out:148
0.013403 http://e3.weblogsinc.com In:130 Out:128
0.013209 http://www.slashfood.com In:165 Out:133
0.012798 http://automoviles.aol.com In:67 Out:103
0.012559 http://www.gadling.com In:165 Out:137
0.012399 http://www.downloadsquad.com/2007/07/16/yousendit-... In:115 Out:138
0.012325 http://digitalmusic.weblogsinc.com In:130 Out:107
0.012235 http://microsoft.weblogsinc.com In:130 Out:128
0.011875 http://www.autoblog.com/category/jaguar In:19 Out:201
0.011861 http://playstation3.weblogsinc.com In:130 Out:128

Figure B.34 – Liste des 10 meilleurs hubs retournée par HITS

0.021927 http://www.ps3fanboy.com In:169 Out:133
0.021216 http://www.nintendowiifanboy.com In:169 Out:143
0.019268 http://www.cardsquad.com In:169 Out:138
0.018737 http://www.scmwire.com In:169 Out:169
0.018538 http://www.blogginggnomedex.com In:169 Out:167
0.018251 http://www.secondlifeinsider.com In:169 Out:143
0.017754 http://www.bbhub.com In:169 Out:138
0.017678 http://www.brianalvey.com In:172 Out:2
0.017548 http://www.thediabetesblog.com In:170 Out:133
0.017466 http://www.bloggingohio.com In:169 Out:148

Figure B.35 – Liste des 10 meilleures autorités retournée par HubAvg

0.0079547 http://meskill.weblogsinc.com In:129 Out:1
0.0071348 http://www.engadgethd.com/2007/07/16/dragons-lair-... In:89 Out:138
0.0070689 http://www.engadgethd.com/2007/07/17/dreambee-... In:89 Out:138
0.0067291 http://www.slashfood.com/2007/07/16/dessert-wine-notes-... In:116 Out:133
0.0067081 http://www.styledash.com In:165 Out:133
0.006681 http://playstation3.weblogsinc.com In:130 Out:128
0.0066792 http://automoviles.aol.com/2007/07/16/cuando-seis-es-... In:64 Out:103
0.0066072 http://www.downloadsquad.com/2007/07/16/yousendit-... In:115 Out:138
0.0065999 http://automoviles.aol.com/2007/07/16/fotos-de-la-... In:64 Out:103
0.0065409 http://automoviles.aol.com/2007/07/16/scion-xd-... In:64 Out:103

Figure B.36 – Liste des 10 meilleurs hubs retournée par HubAvg

0.016255 http://www.ps3fanboy.com In:169 Out:133
0.01587 http://www.nintendowiifanboy.com In:169 Out:143
0.01535 http://www.cardsquad.com In:169 Out:138
0.015001 http://www.blogginggnomedex.com In:169 Out:167
0.01444 http://www.bbhub.com In:169 Out:138
0.014133 http://www.thediabetesblog.com In:170 Out:133
0.014042 http://www.scmwire.com In:169 Out:169
0.01403 http://www.bloggingohio.com In:169 Out:148
0.014002 http://www.secondlifeinsider.com In:169 Out:143
0.013866 http://www.brianalvey.com In:172 Out:2

Figure B.37 – Liste des 10 meilleures autorités retournée par NHITS_10

0.010388 http://stron.frm.pl/wiki.php?title=Jaguar In:0 Out:46
0.010049 http://www.engadget.com In:171 Out:142
0.0098019 http://e3.weblogsinc.com In:130 Out:128
0.0096889 http://www.joox.se/tell-me-more-about-Jaguar In:0 Out:47
0.0096786 http://www.answers.com/topic/jaguar-1 In:1 Out:45
0.009467 http://gps.engadget.com In:161 Out:142
0.0093947 http://www.bloggingohio.com In:169 Out:148
0.0093731 http://en.wikipedia.org/wiki/Jaguar In:16 Out:52
0.0092447 http://automoviles.aol.com In:67 Out:103
0.0091559 http://www.slashfood.com In:165 Out:133

Figure B.38 – Liste des 10 meilleurs hubs retournée par NHITS_10

0.015848 http://www.cardsquad.com In:169 Out:138
0.015729 http://www.ps3fanboy.com In:169 Out:133
0.01507 http://www.nintendowiifanboy.com In:169 Out:143
0.014984 http://www.blogginggnomedex.com In:169 Out:167
0.014663 http://www.bloggingohio.com In:169 Out:148
0.014138 http://www.bbhub.com In:169 Out:138
0.013735 http://www.thediabetesblog.com In:170 Out:133
0.013713 http://www.brianalvey.com In:172 Out:2
0.013569 http://www.secondlifeinsider.com In:169 Out:143
0.013543 http://www.adjab.com In:169 Out:148

Figure B.39 – Liste des 10 meilleures autorités retournée par NHITS_20

0.010816 http://pages.citebite.com/m9u1y2q6voja In:0 Out:115
0.009523 http://stron.frm.pl/wiki.php?title=Jaguar In:0 Out:46
0.0093847 http://www.bloggingohio.com In:169 Out:148
0.0093806 http://e3.weblogsinc.com In:130 Out:128
0.0091675 http://www.engadget.com In:171 Out:142
0.0091098 http://automoviles.aol.com In:67 Out:103
0.0089151 http://www.slashfood.com In:165 Out:133
0.0088749 http://www.joox.se/tell-me-more-about-Jaguar In:0 Out:47
0.0088683 http://www.answers.com/topic/jaguar-1 In:1 Out:45
0.0088151 http://www.gadling.com In:165 Out:137

Figure B.40 – Liste des 10 meilleurs hubs retournée par NHITS_20

0.017085 http://www.ps3fanboy.com In:169 Out:133
0.016809 http://www.nintendowiifanboy.com In:169 Out:143
0.015545 http://www.bloggingggnomedex.com In:169 Out:167
0.01546 http://www.cardsquad.com In:169 Out:138
0.015293 http://www.scmwire.com In:169 Out:169
0.015141 http://www.bbhub.com In:169 Out:138
0.014755 http://www.secondlifeinsider.com In:169 Out:143
0.014345 http://www.brianalvey.com In:172 Out:2
0.014282 http://www.thediabetesblog.com In:170 Out:133
0.014219 http://www.adjab.com In:169 Out:148

Figure B.41 – Liste des 10 meilleures autorités retournée par DocRank_0

0.010437 http://www.bloggingohio.com In:169 Out:148
0.010297 http://e3.weblogsinc.com In:130 Out:128
0.010138 http://www.slashfood.com In:165 Out:133
0.0098911 http://automoviles.aol.com In:67 Out:103
0.0096706 http://www.gadling.com In:165 Out:137
0.009509 http://www.downloadsquad.com/2007/07/16/... In:115 Out:138
0.0094296 http://digitalmusic.weblogsinc.com In:130 Out:107
0.0093253 http://microsoft.weblogsinc.com In:130 Out:128
0.0091567 http://www.autoblog.com/category/jaguar In:19 Out:201
0.0090266 http://playstation3.weblogsinc.com In:130 Out:128

Figure B.42 – Liste des 10 meilleurs hubs retournée par DocRank_0

0.039724 http://www.flickr.com/photos/ableman/sets/72157594421824427 In:46 Out:46
0.034357 http://www.macattorney.com/tutorial.html In:47 Out:65
0.02599 http://www.nalleyjaguar.com In:24 Out:36
0.016891 http://www.ingom.info/Jaguar-clubs.htm In:0 Out:25
0.015936 http://www.autoblog.com/category/jaguar In:19 Out:201
0.015607 http://www.mainjaguar.com In:11 Out:20
0.015461 http://www.putnamjaguar.com In:8 Out:19
0.012633 http://www.oreillynet.com/pub/a/mac/2002/12/27/macosx_... In:5 Out:36
0.010119 http://stron.frm.pl/wiki.php?title=Jaguar In:0 Out:46
0.0095387 http://www.joox.se/tell-me-more-about-Jaguar In:0 Out:47

Figure B.43 – Liste des 10 meilleurs hubs retournée par DocRank_1

0.039053 http://www.flickr.com/photos/ableman/sets/72157594421824427 In:46 Out:46
0.033604 http://www.jaguarmarine.com In:9 Out:1
0.033384 http://www.jaguarpc.com In:30 Out:0
0.021919 http://www.oreillylearning.com In:36 Out:4
0.015963 http://www.jaguarreef.com In:29 Out:2
0.014446 http://www.jaguared.com In:28 Out:2
0.013247 http://www.jiggyjaguar.com In:13 Out:3
0.012326 http://www.jag-lovers.org In:146 Out:4
0.012105 http://www.topblogarea.com/rss/Jaguar.htm In:14 Out:10
0.0115 http://www.mediawiki.org In:61 Out:2

Figure B.44 – Liste des 10 meilleures autorités retournée par DocRank_1.5

0.40391 http://www.nalleyjaguar.com In:24 Out:36
0.12461 http://www.mainjaguar.com In:11 Out:20
0.055922 http://www.oreillynet.com/pub/a/mac/2002/08/23/jaguar_server.html In:5 Out:25
0.047181 http://www.oreillynet.com/pub/a/mac/2002/12/27/macosx_... In:5 Out:36
0.044805 http://www.putnamjaguar.com In:8 Out:19
0.022599 http://www.macattorney.com/tutorial.html In:47 Out:65
0.02162 http://www.ingom.info/Jaguar-clubs.htm In:0 Out:25
0.018284 http://www.flickr.com/photos/ableman/sets/72157594421824427 In:46 Out:46
0.01523 http://koti.phnet.fi/~jagclub/largesites.htm In:0 Out:5
0.012308 http://www.autoblog.com/category/jaguar In:19 Out:201

Figure B.45 – Liste des 10 meilleurs hubs retournée par DocRank_1.5

B.4 Résultats avec le graphe "Washington"

0.014736 http://www.washingtonpost.com In:203 Out:38
0.013647 http://access.wa.gov In:188 Out:0
0.0083479 http://www.redskins.com In:115 Out:8
0.0075494 http://www.wsdot.wa.gov In:104 Out:1
0.0074042 http://www.mediawiki.org In:102 Out:2
0.0066783 http://www.washingtontimes.com In:92 Out:51
0.0066057 http://www.washington.edu In:91 Out:15
0.0066057 http://www.drudgereport.com In:91 Out:10
0.0063879 http://wikimediafoundation.org In:88 Out:1
0.0063153 http://www.senate.gov In:87 Out:0

Figure B.46 – Liste des 10 meilleures autorités retournée par DEG

0.0070412 http://en.wikipedia.org/wiki/Washington,_D.C. In:48 Out:97
0.0070412 http://en.wikipedia.org/wiki/Washington,_DC In:1 Out:97
0.0050087 http://www.washingtonmonthly.com In:77 Out:69
0.0049361 http://www.nfl.com/teams/washingtonredskins/profile?team=WAS In:46 Out:68
0.0048635 http://www.wunderground.com/US/WA In:49 Out:67
0.0047909 http://www.wunderground.com/US/FL/Chipley.html In:1 Out:66
0.004428 http://reddit.com/goto?rss=true&id=t3_61joe In:0 Out:61
0.0043554 http://www.50states.com/washingt.htm In:52 Out:60
0.0042828 http://www.nfl.com In:57 Out:59
0.0042102 http://www.nfl.com/schedules In:5 Out:58

Figure B.47 – Liste des 10 meilleurs hubs retournée par DEG

0.026498 http://www.nfl.com/teams/washingtonredskins/profile?team=WAS In:46 Out:68
0.026226 http://www.nfl.com In:57 Out:59
0.026029 http://reddit.com/goto?rss=true&id=t3_61joe In:0 Out:61
0.025862 http://www.nfl.com/schedules In:5 Out:58
0.024101 http://www.raidernation.com In:0 Out:38
0.023652 http://bengals.com/community/palmer_cornhole06.asp In:0 Out:36
0.023652 http://www.bengals.com In:30 Out:36
0.023652 http://www.cincinnatibengals.com In:0 Out:36
0.023613 http://www.baltimoreravens.com In:49 Out:37
0.023596 http://www.houstontexans.com In:33 Out:36

Figure B.48 – Liste des 10 meilleurs hubs retournée par HITS

0.0036322 http://www.click2houston.com In:2 Out:1
0.0036322 http://www.clickondetroit.com In:4 Out:1
0.0036322 http://www.upenn.edu/almanac In:0 Out:1
0.0036322 http://www.crunchweb.net/87billion In:0 Out:1
0.0036322 http://www.happynews.com In:2 Out:1
0.0036322 http://www.spartacus.schoolnet.co.uk/FWW.htm In:1 Out:1
0.0036322 http://mcadams.posc.mu.edu/home.htm In:1 Out:1
0.0036322 http://www.local6.com In:2 Out:1
0.0036322 http://www.washpostco.com In:41 Out:1
0.0036322 http://www.spartacus.schoolnet.co.uk In:1 Out:1

Figure B.49 – Liste des 10 meilleures autorités retournée par HubAvg

0.082455 http://www.washingtonpost.com In:203 Out:38
0.039894 http://access.wa.gov In:188 Out:0
0.025442 http://www.washingtontimes.com In:92 Out:51
0.019305 http://www.drudgereport.com In:91 Out:10
0.016305 http://www.washington.org In:84 Out:5
0.016184 http://www.dc.gov In:83 Out:2
0.013995 http://www.senate.gov In:87 Out:0
0.012851 http://www.wsdot.wa.gov In:104 Out:1
0.012335 http://www.yahoo.com In:83 Out:0
0.012333 http://www.redskins.com In:115 Out:8

Figure B.50 – Liste des 10 meilleurs hubs retournée par HubAvg

0.013438 http://www.washingtonpost.com In:203 Out:38
0.011162 http://www.mediawiki.org In:102 Out:2
0.010706 http://wikimediafoundation.org In:88 Out:1
0.0089608 http://wikimediafoundation.org/wiki/Privacy_policy In:65 Out:1
0.0082013 http://access.wa.gov In:188 Out:0
0.0081715 http://www.redskins.com In:115 Out:8
0.0080719 http://wikimediafoundation.org/wiki/Donate In:55 Out:0
0.0075863 http://www.wikimediafoundation.org In:49 Out:1
0.0073304 http://wikimediafoundation.org/wiki/Deductibility_of_donations In:46 Out:1
0.0069909 http://www.washpostco.com In:41 Out:1

Figure B.51 – Liste des 10 meilleures autorités retournée par NHITS_10

0.011486 http://www.nfl.com/teams/washingtonredskins/profile?team=WAS In:46 Out:68
0.01114 http://reddit.com/goto?rss=true&id=t3_61joe In:0 Out:61
0.010957 http://en.wikipedia.org/wiki/Washington,_D.C. In:48 Out:97
0.010957 http://en.wikipedia.org/wiki/Washington,_DC In:1 Out:97
0.010915 http://www.nfl.com In:57 Out:59
0.010714 http://www.nfl.com/schedules In:5 Out:58
0.0067411 http://dc.indymedia.org In:28 Out:34
0.0067207 http://www.thewashingtonpost.com In:1 Out:41
0.0067207 http://www.washintonpost.com In:0 Out:41
0.0067207 http://www.washingtonpos.com In:0 Out:41

Figure B.52 – Liste des 10 meilleurs hubs retournée par NHITS_10

0.012921 http://www.washingtonpost.com In:203 Out:38
0.012552 http://access.wa.gov In:188 Out:0
0.010709 http://www.mediawiki.org In:102 Out:2
0.010192 http://wikimediafoundation.org In:88 Out:1
0.0094664 http://www.redskins.com In:115 Out:8
0.0084564 http://wikimediafoundation.org/wiki/Privacy_policy In:65 Out:1
0.0079465 http://www.washington.edu In:91 Out:15
0.0076366 http://wikimediafoundation.org/wiki/Donate In:55 Out:0
0.0073259 http://www.senate.gov In:87 Out:0
0.0071662 http://www.wikimediafoundation.org In:49 Out:1

Figure B.53 – Liste des 10 meilleures autorités retournée par NHITS_20

0.018701 http://www.wunderground.com/US/WA In:49 Out:67
0.018561 http://www.wunderground.com/US/FL/Chipley.html In:1 Out:66
0.013171 http://en.wikipedia.org/wiki/Washington,_D.C. In:48 Out:97
0.013171 http://en.wikipedia.org/wiki/Washington,_DC In:1 Out:97
0.0076559 http://www.nfl.com/teams/washingtonredskins/profile?team=WAS In:46 Out:68
0.0074295 http://reddit.com/goto?rss=true&id=t3_61joe In:0 Out:61
0.0072754 http://www.nfl.com In:57 Out:59
0.0071505 http://www.nfl.com/schedules In:5 Out:58
0.0057643 http://en.wikipedia.org/wiki/Washington In:60 Out:55
0.0057643 http://en.wikipedia.org/wiki/washington In:3 Out:55

Figure B.54 – Liste des 10 meilleurs hubs retournée par NHITS_20

0.0088646 http://www.washingtonpost.com In:203 Out:38
0.0083662 http://www.redskins.com In:115 Out:8
0.0075564 http://access.wa.gov In:188 Out:0
0.0064208 http://www.seahawks.com In:50 Out:0
0.0062531 http://www.baltimoreravens.com In:49 Out:37
0.0062292 http://www.sf49ers.com In:36 Out:0
0.0062052 http://www.denverbroncos.com In:36 Out:1
0.0062004 http://www.steelers.com In:39 Out:0
0.0061908 http://www.chicagobears.com In:36 Out:1
0.0061812 http://www.neworleanssaints.com In:35 Out:1

Figure B.55 – Liste des 10 meilleures autorités retournée par DocRank_0

0.003744 http://washington.lap.hu In:0 Out:30
0.0029748 http://usa.allepaginas.nl In:0 Out:35
0.0029386 http://www.washlaw.edu/uslaw/states/Washington In:0 Out:30
0.0029024 http://amerika.startje.com In:0 Out:35
0.0028196 http://www.statelocalgov.net/state-wa.cfm In:3 Out:45
0.0026127 http://javascript-reference.info/java-directory/Regional_United_... In:0 Out:25
0.0024799 http://reddit.com/goto?rss=true&id=t3_61joe In:0 Out:61
0.0024627 http://govdocs.evergreen.edu/wastate/walpha.html In:0 Out:31
0.0024471 http://www.nfl.com/teams/washingtonredskins/profile?team=WAS In:46 Out:68
0.0024247 http://www.nfl.com In:57 Out:59

Figure B.56 – Liste des 10 meilleurs hubs retournée par DocRank_0

0.039634 http://www.washingtonmonthly.com In:77 Out:69
0.027226 http://en.wikipedia.org/wiki/Washington,_D.C. In:48 Out:97
0.027226 http://en.wikipedia.org/wiki/Washington,_DC In:1 Out:97
0.020768 http://www.wunderground.com/US/WA In:49 Out:67
0.020138 http://www.wunderground.com/US/FL/Chipley.html In:1 Out:66
0.016153 http://www.washingtontimes.com In:92 Out:51
0.013471 http://www.50states.com/washingt.htm In:52 Out:60
0.013156 http://sports.espn.go.com/nba/clubhouse?team=WAS In:15 Out:30
0.012711 http://sports.espn.go.com/nfl/clubhouse?team=was In:45 Out:30
0.011218 http://www.travel-in-wa.com In:59 Out:26

Figure B.57 – Liste des 10 meilleurs hubs retournée par DocRank_1

0.01564 http://www.worldtimeserver.com/current_time_in_US-WA.aspx In:48 Out:8
0.015385 http://geo.craigslist.org/iso/us/wa In:48 Out:1
0.014394 http://www.imdb.com/name/nm0000243 In:48 Out:1
0.014322 http://www.flickr.com/photos/tags/washington In:48 Out:8
0.013628 http://www.washingtoncountytn.com In:50 Out:3
0.01325 http://www.missingkids.com/precreate/WA.html In:53 Out:0
0.013225 http://www.washingtonwine.org In:56 Out:4
0.013087 http://www.wcs.k12.va.us In:43 Out:3
0.013052 http://www.washington.org In:84 Out:5
0.012776 http://www.wunderground.com/US/WA In:49 Out:67

Figure B.58 – Liste des 10 meilleures autorités retournée par DocRank_1.5

0.070656 http://www.washingtonmonthly.com In:77 Out:69
0.025246 http://www.washingtontimes.com In:92 Out:51
0.024747 http://en.wikipedia.org/wiki/Washington,_D.C. In:48 Out:97
0.024747 http://en.wikipedia.org/wiki/Washington,_DC In:1 Out:97
0.021978 http://sports.espn.go.com/nba/clubhouse?team=WAS In:15 Out:30
0.020996 http://sports.espn.go.com/nfl/clubhouse?team=was In:45 Out:30
0.019403 http://www.wunderground.com/US/WA In:49 Out:67
0.018673 http://www.50states.com/washingt.htm In:52 Out:60
0.018318 http://www.travel-in-wa.com In:59 Out:26
0.018241 http://www.wunderground.com/US/FL/Chipley.html In:1 Out:66

Figure B.59 – Liste des 10 meilleurs hubs retournée par DocRank_1.5

Bibliographie

[Aggarwal and Wang 10] C. C. Aggarwal and H. Wang. A Survey of Clustering Algorithms for Graph Data. *Managing and Mining Graph Data*. A. K. Elmagarmid, Springer US, 40: 275-301, 2010.

[Agresti 07] A. Agresti. *An Introduction to Categorical Data Analysis, 2nd Edition*. Wiley, 2007.

[Alba 73] R. D. Alba. A graph-theoretic definition of a sociometric clique. *The Journal of Mathematical Sociology*, 3(1):113 - 126, 1973.

[Albert et al. 99] R. Albert, H. Jeong and A.-L. Barabasi. Internet: Diameter of the World-Wide Web. *Nature*, 401(6749):130-131, 1999.

[Amsler 72] R. Amsler. Application of citation-based automatic classification, The University of Texas at Austin, Linguistics Research Center, 1972.

[Anderson et al. 92] C. J. Anderson, S. Wasserman and K. Faust. Building stochastic blockmodels. *Social Networks*, 14(1-2):137-161, 1992.

[Baeza-Yates and Ribeiro-Neto 99] R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. Addison Wesley, 1999.

[Baldi et al. 03] P. Baldi, P. Frasconi and P. Smyth. *Modeling the Internet and the Web: Probabilistic Methods and Algorithms*. Wiley, 2003.

[Barabasi 09] A.-L. Barabasi. Scale-Free Networks: A Decade and Beyond. *Science*, 325(5939):412-413, 2009.

[Barnes 82] E. R. Barnes. An Algorithm for Partitioning the Nodes of a Graph. *SIAM Journal on Algebraic and Discrete Methods*, 3(4):541-550, 1982.

[Bennouas 05] T. Bennouas. *Modélisation de parcours du web et calcul de communautés par émergence*. Thèse de doctorat, Université Montpellier II, 2005.

[Bharat and Henzinger 98] K. Bharat and M. R. Henzinger. Improved algorithms for topic distillation in a hyperlinked environment. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, Melbourne, Australia, ACM, pages 104-111, 1998.

[Bishop 07] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2007.

[Bonacich 72] P. Bonacich. Factoring and weighting approaches to status scores and clique identification. *Journal of Mathematical Sociology*, 2(1):113-120, 1972.

[Bonacich 07] P. Bonacich. Some unique properties of eigenvector centrality. *Social Networks*, 29(4):555-564, 2007.

[Borgatti and Everett 06] S. P. Borgatti and M. G. Everett. A Graph-theoretic perspective on centrality. *Social Networks*, 28(4):466-484, 2006.

[Bornholdt and Schuster 03] S. Bornholdt and H. G. Schuster. *Handbook of Graphs and Networks: From the Genome to the Internet*. Wiley, 2003.

[Borodin et al. 01] A. Borodin, G. O. Roberts, J. S. Rosenthal and P. Tsaparas. Finding authorities and hubs from link structures on the World Wide Web. In *Proceedings of the 10th international conference on World Wide Web*, Hong Kong, Hong Kong, ACM, pages, 2001.

[Borodin et al. 05] A. Borodin, G. O. Roberts, J. S. Rosenthal and P. Tsaparas. Link analysis ranking: algorithms, theory, and experiments. *ACM Trans. Internet Technol.*, 5(1):231-297, 2005.

[Bouklit 06] M. Bouklit. *Autour du graphe du web : Modélisations probabilistes de l'internaute et détection de structures de communauté*. Thèse de doctorat, Université Montpellier II, 2006.

[Brin and Page 98] S. Brin and L. Page. The anatomy of a large-scale hypertextual Web search engine. *Comput. Netw. ISDN Syst.*, 30(1-7):107-117, 1998.

[Broder et al. 00] A. Broder, R. Kumar, F. Maghoul, P. Raghavan, S. Rajagopalan, R. Stata, A. Tomkins and J. Wiener. Graph structure in the Web. *Comput. Netw.*, 33(1-6):309-320, 2000.

[Brown and Fuchs 83] M. B. Brown and C. Fuchs. On maximum likelihood estimation in sparse contingency tables. *Computational Statistics & Data Analysis*, 1:3-15, 1983.

[Calado et al. 03] P. Calado, M. Cristo, E. Moura, N. Ziviani, B. Ribeiro-Neto and M. A. Goncalves. Combining link-based and content-based methods for web document classification. In *Proceedings of the12th international conference on Information and knowledge management*, New Orleans, LA, USA, ACM, pages 394-401, 2003.

[Caldarelli 07] G. Caldarelli. *Scale-Free Networks: Complex Webs in Nature and Technology*. Oxford University Press, 2007.

[Chakrabarti 02] S. Chakrabarti. *Mining the Web: Discovering Knowledge from Hypertext Data*. Morgan Kaufmann, 2002.

[Chakrabarti et al. 01] S. Chakrabarti, M. Joshi and V. Tawde. Enhanced topic distillation using text, markup tags, and hyperlinks. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, New Orleans, Louisiana, United States, ACM, pages 208-216, 2001.

[Chen et al. 07] P. Chen, H. Xie, S. Maslov and S. Redner. Finding scientific gems with Google's PageRank algorithm. *Journal of Informetrics*, 1(1):8-15, 2007.

[Chikhi et al. 07] N. F. Chikhi, B. Rothenburger and N. Aussenac-Gilles. A Comparison of Dimensionality Reduction Techniques for Web Structure Mining. In *IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, Silicon Valley, California (USA), IEEE Computer Society, pages 116-119, 2007.

[Chikhi et al. 08a] N. F. Chikhi, B. Rothenburger and N. Aussenac-Gilles. Authoritative Documents Identification based on Nonnegative Matrix Factorization. In *IEEE International Conference on Information Reuse and Integration (IRI)*, Las Vegas, Nevada (USA), IEEE, pages 262-267, 2008.

[Chikhi et al. 08b] N. F. Chikhi, B. Rothenburger and N. Aussenac-Gilles. Combining Link and Content Information for Scientific Topics Discovery. In *IEEE International Conference on Tools with Artificial Intelligence (ICTAI)*, Dayton, Ohio (USA), IEEE Computer Society, pages 211-214, 2008.

[Chikhi et al. 09] N. F. Chikhi, B. Rothenburger and N. Aussenac-Gilles. Community Structure Identification: A Probabilistic Approach. In *International Conference on Machine Learning and Applications (ICMLA)*, Miami, Florida (USA), IEEE Computer Society, pages 125-130, 2009.

[Chikhi et al. 10] N. F. Chikhi, B. Rothenburger and N. Aussenac-Gilles. Une approche probabiliste pour l'identification de structures de communautés. In *Conférence Internationale Francophone sur l'Extraction et la Gestion des Connaissances (EGC)*, Hammamet (Tunisie), Cépaduès Editions, pages 175-180, 2010.

[Cohn and Chang 00] D. Cohn and H. Chang. Learning to Probabilistically Identify Authoritative Documents. In *Proceedings of the 7th International Conference on Machine Learning*, Morgan Kaufmann Publishers Inc., pages 167-174, 2000.

[Cohn and Hofmann 01] D. Cohn and T. Hofmann. The missing link - A probabilistic model of document content and hypertext connectivity. In *Proceedings of the 13th Neural Information Processing Systems Conference*, Vancouver, British Columbia, Canada, pages 430–436, 2001.

[Cook and Holder 06] D. J. Cook and L. B. Holder. *Mining Graph Data*. Wiley, 2006.

[Dahinden et al. 07] C. Dahinden, G. Parmigiani, M. Emerick and P. Buhlmann. Penalized likelihood for sparse contingency tables with an application to full-length cDNA libraries. *BMC Bioinformatics*, 8(1):476, 2007.

[Daudin et al. 08] J. J. Daudin, F. Picard and S. Robin. A mixture model for random graphs. *Statistics and Computing*, 18(2):173-183, 2008.

[Dempster et al. 77] A. P. Dempster, N. M. Laird and D. B. Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society*, 39(1):1-38, 1977.

[Dhillon and Guan 03] I. S. Dhillon and Y. Guan. Information theoretic clustering of sparse cooccurrence data. In *Third IEEE International Conference on Data Mining*, pages 517-520, 2003.

[Dhillon et al. 07] I. S. Dhillon, Y. Guan and B. Kulis. Weighted Graph Cuts without Eigenvectors A Multilevel Approach. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(11):1944-1957, 2007.

[Ding et al. 02] C. Ding, X. He, P. Husbands, H. Zha and H. D. Simon. PageRank, HITS and a unified framework for link analysis. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, Tampere, Finland, ACM, pages, 2002.

[Donetti and Munoz 04] L. Donetti and M. A. Munoz. Detecting network communities: a new systematic and efficient algorithm. *Journal of Statistical Mechanics: Theory and Experiment*, 2004(10):P10012, 2004.

[Dourisboure et al. 07] Y. Dourisboure, F. Geraci and M. Pellegrini. Extraction and classification of dense communities in the web. In *Proceedings of the 16th international conference on World Wide Web*, Banff, Alberta, Canada, ACM, pages 461-470, 2007.

[Drost et al. 06] I. Drost, S. Bickel and T. Scheffer. Discovering Communities in Linked Data by Multi-view Clustering. In *From Data and Information Analysis to Knowledge Engineering*. Springer, pages 342-349, 2006.

[Farahat et al. 05] A. Farahat, T. LoFaro, J. C. Miller, G. Rae and L. A. Ward. Authority Rankings from HITS, PageRank, and SALSA: Existence, Uniqueness, and Effect of Initialization. *SIAM J. Sci. Comput.*, 27(4):1181-1201, 2005.

[Fiala et al. 08] D. Fiala, F. Rousselot and K. Ježek. PageRank for bibliographic networks. *Scientometrics*, 76(1):135-158, 2008.

[Flake et al. 00] G. W. Flake, S. Lawrence and C. L. Giles. Efficient identification of Web communities. In *Proceedings of the 6th ACM SIGKDD international conference on Knowledge discovery and data mining*, Boston, Massachusetts, United States, ACM, pages 150-160, 2000.

[Flake et al. 04] G. W. Flake, K. Tsioutsiouliklis and L. Zhukov. Web Communities: Bibliometric, Spectral, and Flow. In *Web Dynamics: Adapting To Change In Content, Size, Topology And Use*. Springer, pages 45-68, 2004.

[Fortunato 10] S. Fortunato. Community detection in graphs. *Physics Reports*, 486(3-5):75-174, 2010.

[Fortunato 08] S. Fortunato, M. Boguñá, A. Flammini and F. Menczer. Approximating PageRank from In-Degree. *Algorithms and Models for the Web-Graph*. W. Aiello, A. Broder, J. Janssen and E. Milios, Springer Berlin / Heidelberg. **4936:** 59-71. 2008.

[Fortunato and Barthélemy 07] S. Fortunato and M. Barthélemy. Resolution limit in community detection. *Proceedings of the National Academy of Sciences*, 104(1):36-41, 2007.

[Freeman 79] L. C. Freeman. Centrality in social networks conceptual clarification. *Social Networks*, 1(3):215-239, 1979.

[Freeman 91] L. Freeman. Centrality in valued graphs: A measure of betweenness based on network flow. *Social Networks*, 13(2):141-154, 1991.

[Garfield 70] E. Garfield. Citation Indexing for Studying Science. *Nature*, 227(5259):669-671, 1970.

[Garfield 72] E. Garfield. Citation analysis as a tool in journal evaluation. *Science*, 178(60):471-479, 1972.

[Getoor and Diehl 05] L. Getoor and C. P. Diehl. Link mining: a survey. *SIGKDD Explor. Newsl.*, 7(2):3-12, 2005.

[Girvan and Newman 02] M. Girvan and M. E. J. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences of the United States of America*, 99(12):7821-7826, 2002.

[Golub and Van Loan 96] G. H. Golub and C. F. Van Loan. *Matrix Computations, 3rd edition*. The Johns Hopkins University Press, 1996.

[Guillaume 04] J.-L. Guillaume. *Analyse statistique et modélisation des grands réseaux d'interactions*. Thèse de doctorat, Université Denis Diderot - Paris VII, 2004.

[Halkidi et al. 01] M. Halkidi, Y. Batistakis and M. Vazirgiannis. On Clustering Validation Techniques. *J. Intell. Inf. Syst.*, 17(2-3):107-145, 2001.

[Hastings 06] M. B. Hastings. Community detection as an inference problem. *Physical Review E*, 74(3):035102, 2006.

[Hofman and Wiggins 08] J. M. Hofman and C. H. Wiggins. Bayesian Approach to Network Modularity. *Physical Review Letters*, 100(25):258701, 2008.

[Hofmann 99] T. Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, Berkeley, California, United States, ACM, pages 50-57, 1999.

[Hofmann 01] T. Hofmann. Unsupervised learning by probabilistic latent semantic analysis. *Mach. Learn.*, 42(1/2):177-196, 2001.

[Jain 10] A. K. Jain. Data clustering: 50 years beyond K-means. *Pattern Recognition Letters*, 31(8):651-666, 2010.

[Jain and Dubes 98] A. K. Jain and R. C. Dubes. *Algorithms for Clustering Data*. Prentice Hall College Div, 1988.

[Jain et al. 99] A. K. Jain, M. N. Murty and P. J. Flynn. Data clustering: a review. *ACM Comput. Surv.*, 31(3):264-323, 1999.

[Janssens 07] F. Janssens. *Clustering of scientific fileds by integrating text mining and bibliometrics*. PhD Thesis, Katholieke Universiteit Leuven, 2007.

[Jo et al. 07] Y. Jo, C. Lagoze and C. L. Giles. Detecting research topics via the correlation between graphs and texts. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, San Jose, California, USA, ACM, pages, 2007.

[Kessler 63] M. M. Kessler. Bibliographic coupling between scientific papers. *American Documentation*, 14(10):10-25, 1963.

[Kleinberg 98] J. Kleinberg. Authoritative sources in a hyperlinked environment. In *Proceedings of the ninth annual ACM-SIAM symposium on Discrete algorithms*, San Francisco, California, United States, Society for Industrial and Applied Mathematics, pages, 1998.

[Kleinberg 99a] J. Kleinberg. Authoritative sources in a hyperlinked environment. *J. ACM*, 46(5):604-632, 1999.

[Kleinberg 99b] J. Kleinberg, R. Kumar, P. Raghavan, S. Rajagopalan and A. Tomkins. The Web as a Graph: Measurements, Models, and Methods. In *Computing and Combinatorics*. Springer, 1627: 1-17, 1999.

[Koschützki et al. 05] D. Koschützki, K. A. Lehmann, L. Peeters, S. Richter, D. Tenfelde-Podehl and O. Zlotowski. Centrality Indices. In *Network Analysis*. Springer Berlin / Heidelberg. 3418: 16-61, 2005.

[Kumar et al. 06] R. Kumar, P. Raghavan, S. Rajagopalan and A. Tomkins. Core algorithms in the CLEVER system. *ACM Trans. Internet Technol.*, 6(2):131-152, 2006.

[Lancichinetti and Fortunato 09] A. Lancichinetti and S. Fortunato. Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities. *Physical Review E*, 80(1):016118, 2009.

[Langville and Meyer 05] A. Langville and C. Meyer. A Survey of Eigenvector Methods for Web Information Retrieval. *SIAM Rev.*, 47(1):135-161, 2005.

[Langville and Meyer 06] A. Langville and C. Meyer. *Google's PageRank and Beyond: The Science of Search Engine Rankings*. Princeton University Press, 2006.

[Latouche et al. 10] P. Latouche, E. Birmelé and C. Ambroise. Bayesian Methods for Graph Clustering. In *Advances in Data Analysis, Data Handling and Business Intelligence*. Springer Berlin Heidelberg, pages 229-239, 2010.

[Lee and Seung 99] D. D. Lee and H. S. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401:788-791, 1999.

[Lee and Seung 01] D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. In *Proceedings of the 13th Neural Information Processing Systems Conference*, pages, 2001.

[Leicht and Newman 08] E. A. Leicht and M. E. J. Newman. Community Structure in Directed Networks. *Physical Review Letters*, 100(11):118703, 2008.

[Lempel and Moran 00] R. Lempel and S. Moran. The stochastic approach for link-structure analysis (SALSA) and the TKC effect. *Comput. Netw.*, 33(1-6):387-401, 2000.

[Liu 06] B. Liu. *Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data*. Springer, 2006.

[Lu and Getoor 03] Q. Lu and L. Getoor. Link-based Classification. In *Proceedings of the 20th International Conference on Machine Learning (ICML)*, Washington, DC, USA, 2003.

[Luce and Perry 49] R. Luce and A. Perry. A method of matrix analysis of group structure. *PSYCHOMETRIKA*, 14(2):95-116, 1949.

[Ma et al. 08] N. Ma, J. Guan and Y. Zhao. Bringing PageRank to the citation analysis. *Information Processing & Management*, 44(2):800-810, 2008.

[MacKay 02] D. J. C. MacKay. *Information Theory, Inference & Learning Algorithms*. Cambridge University Press, 2002.

[Macqueen 67] J. B. Macqueen. Some methods of classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, pages 281-297, 1967.

[Manning et al. 08] C. D. Manning, P. Raghavan and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press 2008.

[Maslov and Redner 08] S. Maslov and S. Redner. Promise and Pitfalls of Extending Google's PageRank Algorithm to Citation Networks. *The Journal of Neuroscience*, 28(44):11103-11105, 2008.

[McLachan and Krishnan 97] G. McLachan and T. Krishnan. *EM Algorithm and Extensions*. Wiley, 1997.

[Modha and Spangler 00] D. S. Modha and W. S. Spangler. Clustering hypertext with applications to web searching. In *Proceedings of the eleventh ACM on Hypertext and hypermedia*, San Antonio, Texas, United States, ACM, pages 143-152, 2000.

[Mokken 79] R. J. Mokken. Cliques, clubs and clans. *Quality & Quantity*, 13(2):161-173, 1979.

[Newman 04] M. E. J. Newman. Fast algorithm for detecting community structure in networks. *Physical Review E*, 69(6):066133, 2004.

[Newman 04a] M. E. J. Newman. Detecting community structure in networks. *The European Physical Journal B - Condensed Matter and Complex Systems*, 38(2):321-330, 2004.

[Newman 06] M. E. J. Newman. Modularity and community structure in networks. *Proceedings of the National Academy of Sciences*, 103(23):8577-8582, 2006.

[Newman and Girvan 04] M. E. J. Newman and M. Girvan. Finding and evaluating community structure in networks. *Physical Review E*, 69(2):026113, 2004.

[Newman and Leicht 07] M. E. J. Newman and E. A. Leicht. Mixture Models and Exploratory Analysis in Networks. *Proceedings of the National Academy of Sciences of the United States of America*, 104(23):9564-9569, 2007.

[Nigam et al. 00] K. Nigam, A. K. McCallum, S. Thrun and T. Mitchell. Text Classification from Labeled and Unlabeled Documents using EM. *Mach. Learn.*, 39(2-3):103-134, 2000.

[Nowicki and Snijders 01] K. Nowicki and T. A. B. Snijders. Estimation and Prediction for Stochastic Blockstructures. *Journal of the American Statistical Association*, 96(455):1077-1087, 2001.

[Pons and Latapy 05] P. Pons and M. Latapy. Computing Communities in Large Networks Using Random Walks. *Computer and Information Sciences - ISCIS 2005*. P. Yolum, T. Güngör, F. Gürgen and C. Özturan, Springer Berlin / Heidelberg. **3733:** 284-293. 2005.

[Pons 07] P. Pons. *Détection de communautés dans les grands graphes de terrain*. Thèse de doctorat, Université Paris 7, 2007.

[Popescul et al. 01] A. Popescul, L. H. Ungar, D. M. Pennock and S. Lawrence. Probabilistic Models for Unified Collaborative and Content-Based Recommendation in Sparse-Data Environments. In *Proceedings of the 17th Conference in Uncertainty in Artificial Intelligence*, Morgan Kaufmann Publishers Inc., pages, 2001.

[Porter et al. 09] M. A. Porter, J.-P. Onnela and P. J. Mucha. Communities in Networks. *Notices of the American Mathematical Society*, 56(9):1082-1097, 2009.

[Price 65] D. J. d. S. Price. Networks of Scientific Papers. *Science*, 149(3683):510-515, 1965.

[Radicchi et al. 04] F. Radicchi, C. Castellano, F. Cecconi, V. Loreto and D. Parisi. Defining and identifying communities in networks. *Proceedings of the National Academy of Sciences of the United States of America*, 101(9):2658-2663, 2004.

[Redner 98] S. Redner. How popular is your paper? An empirical study of the citation distribution. *The European Physical Journal B - Condensed Matter and Complex Systems*, 4(2):131-134, 1998.

[Ren et al. 09] W. Ren, G. Yan, X. Liao and L. Xiao. Simple probabilistic algorithm for detecting community structure. *Physical Review E*, 79(3):036111, 2009.

[Rigouste 06] L. Rigouste. *Méthodes probabilistes pour l'analyse exploratoire de données textuelles*. Thèse de doctorat, École Nationale Supérieure des Télécommunications, 2006.

[Schaeffer 07] S. E. Schaeffer. Graph clustering. *Computer Science Review*, 1(1):27-64, 2007.

[Scharnhorst and Thelwall 05] A. Scharnhorst and M. Thelwall. Citation and hyperlink networks. *Current Science*, 89(9):1518-1524, 2005.

[Scott 00] J. P. Scott. *Social Network Analysis: A Handbook*. Sage Publications Ltd, 2000.

[Seidman and Foster 78] S. B. Seidman and B. L. Foster. A graph-theoretic generalization of the clique concept. *The Journal of Mathematical Sociology*, 6(1):139 - 154, 1978.

[Shi and Malik 97] J. Shi and J. Malik. Normalized Cuts and Image Segmentation. In *Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)*, IEEE Computer Society, pages, 1997.

[Simonoff 98] J. S. Simonoff. *Smoothing Methods in Statistics*. Springer, 1998.

[Small 73] H. Small. Co-citation in the scientific literature: A new measure of the relationship between two documents. *Journal of the American Society for Information Science*, 24(4):265-269, 1973.

[Strehl 02] A. Strehl. *Relationship-based clustering and cluster ensembles for high-dimensional data mining*. PhD Thesis, The University of Texas at Austin, 2002.

[Tan et al. 05] P.-N. Tan, M. Steinbach and V. Kumar (2005). Cluster Analysis: Basic Concepts and Algorithms. In *Introduction to Data Mining*, Addison Wesley, 2005.

[Tsaparas 03] P. Tsaparas. *Link Analysis Ranking*. PhD Thesis, University of Toronto, 2003.

[Utsugi 97] A. Utsugi. Hyperparameter selection for self-organizing maps. *Neural Comput.*, 9(3):623-635, 1997.

[Wang and Kitsuregawa 02] Y. Wang and M. Kitsuregawa. On Combining Link and Contents Information for Web Page Clustering. In *Database and Expert Systems Applications*. Springer, pages 487-566, 2002.

[Wasserman and Faust 94] S. Wasserman and K. Faust. *Social Network Analysis: Methods and Applications*. Cambridge University Press, 1994.

[West 00] D. B. West. *Introduction to Graph Theory*. Prentice Hall, 2000.

[Wu et al. 08] X. Wu, V. Kumar, J. Ross Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. McLachlan, A. Ng, B. Liu, P. Yu, Z.-H. Zhou, M. Steinbach, D. Hand and D. Steinberg. Top 10 algorithms in data mining. *Knowledge and Information Systems*, 14(1):1-37, 2008.

[Xu and Wunsch 08] R. Xu and D. Wunsch. *CLUSTERING*. IEEE Press, 2008.

[Yoon and Park 04] B. Yoon and Y. Park. A text-mining-based patent network: Analytical tool for high-technology trend. *The Journal of High Technology Management Research*, 15(1):37-50, 2004.

[Zhang et al. 05] Y. Zhang, J. X. Yu and J. Hou. *Web Communities: Analysis and Construction*. Springer, 2005.

[Zhou et al. 07] Z.-H. Zhou, H. Li, Q. Yang, L. Yen, F. Fouss, C. Decaestecker, P. Francq and M. Saerens. Graph Nodes Clustering Based on the Commute-Time Kernel. *Advances in Knowledge Discovery and Data Mining*, Springer Berlin / Heidelberg. **4426:** 1037-1045. 2007.

[URL 1] http://www.cs.umd.edu/~sen/lbc-proj/LBC.html

[URL 2] http://research.microsoft.com/en-us/um/people/minka/software/fastfit/